K-MER BASED DATA STRUCTURES AND HEURISTICS FOR MICROBES AND CANCER

A Dissertation
Presented to
The Academic Faculty

by

Cai Huang

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Biology

Georgia Institute of Technology
August 2018

K-MER BASED DATA STRUCTURES AND HEURISTICS FOR MICROBES AND CANCER

Approved by:

Dr. Fredrik Vannberg, Advisor
School of Biology
Georgia Institute of Technology

Dr. Robert Dickson
School of Chemistry & Biochemistry
Georgia Institute of Technology

Dr. John McDonald
School of Biology
Georgia Institute of Technology

Dr. Jung H. Choi
School of Biology
Georgia Institute of Technology

Dr. King Jordan
School of Biology
Georgia Institute of Technology

Date Approved: [May 07, 2018]

To my family

Xue Yu

Alice Huang

Anica Huang

For their care, understanding and great support

ACKNOWLEDGEMENTS

I would like to give my deepest thanks to my committee members for helping me to complete this dissertation.

Foremost, I would like to express my sincere gratitude to my advisor, Dr. Fredrik Vannberg, for the tremendous and continuous support of my Ph.D. study and research, for his motivation, immense knowledge and rigorous teaching and research spirit. His guidance helped me through my research and the writing of this dissertation. I could not have imagined having a better advisor and mentor for my Ph.D. study. It has been a rewarding experience, which will prove to be a fortune to my work in the future.

Besides my advisor, I would like to thank the rest of my dissertation committee: Dr. John McDonald, Dr. King Jordan, Dr. Robert Dickson and Dr. Jung H. Choi, who kindly agreed to serve on my dissertation committee and have given me continuous help, encouragement and insightful comments. It would not have been possible to conduct this research without their treasured support.

I would also like to thank other professors from the Computational Science and Engineering Department for educating me in various courses. I would especially like to mention Dr. Le Song who supported and helped me during my graduate studies.

Furthermore, I would be remiss not show my deepest appreciation to my friends, Peter Audano, Shengyun Peng, Yasvanth Kulasekarapandian and Shashidhar

Ravishankar. Peter is a Bioinformatics Specialist at UW Medicine, who is the smartest and most meticulous person I have ever met. I believe he will be a great Bioinformatics professor. I will miss laughing and chatting in the department with Shashi and Yesh, who supported me unconditionally.

Also, I like to mention Roman Mezencev, who did data curation of microarray data for NCI-60 cell lines and ovarian cancer patients. Evan A. Clayton, who organized patients RNA-seq data from The Cancer Genome Atlas (TCGA) database.

Last but not the least, I want to thank everyone who has helped me throughout my graduate life, especially my immediate family to whom this dissertation is dedicated to. I would like to thank my dearest parents, who provide me with solid support. My wife Xue Yu, who will also get her Ph.D. in Statistics this summer. We take care of each other, and we've grown together. It is a wonderful achievement for our family. I would also like to express many thanks to all my family members in China, for supporting me spiritually throughout my life.

This work was supported in part by an NIH R01 grant from Biomedical Imaging and Bioengineering entitled "QuBBD: Viral Evolution and Spread of Infectious Diseases in Complex Networks" (R01 EB025022; PI: Vannberg). The major goals of this project is to investigate spatiotemporal outbreaks of HIV and HCV using novel heuristics and algorithms to benefit public health agencies such as the Centers for Disease Control.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

BWT	Burrows-Wheeler Transform
CML	Chronic Myelogenous Leukemia
FN	False negative
FP	False positive
FPGA	Field-Programmable Gate Array
GEO	Gene Expression Omnibus
GI-50	Growth inhibition 50
GWAS	Genome-Wide Association Studies
hPCA	hierarchical Principal Component Analysis
H(s)	Shannon entropy
k-matrix	k-mer base matrix
LOOCV	Leave One Out Cross-Validation
LSVM	Linear Support Vector Machine
ML	Machine Learning
MSA	Multiple Sequence Alignment
NCI-60	National Cancer Institute panel of 60 human cancer cell lines
NGS	Next-generation Sequencing
OC	Ovarian Cancer
PCA	Principal Component Analysis
QTL	Quantitative Trail Loci
RFE	Recursive Feature Elimination

RNA-seq	Whole-transcriptome shotgun sequencing
SVM	Support Vector Machine
TCGA	The Cancer Genome Atlas
TK	Tyrosine Kinase
TN	True negative
TP	True Positive
UQ-FPKMs	Upper-Quartile Normalized Fragments Per Kilobase per Million mapped reads
$x \wedge y$	Conjunction of x and y
$x \vee y$	Disjunction of x and y
$\neg x$	Negation of x
$x \oplus y$	Exclusive or of x and y
τ	k-mer occupancy

SUMMARY

Recent technological advances allow for high throughput profiling of biological systems in a cost-efficient manner. The low cost of data generation is leading us to the "big data" era within the broader field of biomedicine. The availability of biology-based big data provides unprecedented opportunities but also raises new challenges for data mining and analysis. Machine learning algorithms have demonstrated their power of increasing efficiency and accuracy in bioinformatics analysis and recent efforts using machine learning to analyze biological data have benefited the computational biology community. Although progress is being made to date there are still many algorithms and heuristics that are not being shared in their entirety to scientific community. This dissertation helps to reverse this by presenting a truly open source platform to perform un-supervised genomic clustering and supervised clustering of cancer patient drug response.

I started by explore the classic clustering algorithms and applying these algorithms to molecular surveillance and also studying infectious disease dynamics. Traditionally, computational methods for this type of analysis are based on gapped alignments to a reference followed by the construction of a phylogenetic tree. In molecular epidemiology, transmission clusters are usually inferred by identifying grouped samples within a phylogenetic tree that can help delineate which viruses were likely transmitted from person to person (i.e. contact tracing). In the past decade, great progress has been achieved in the field of phylodynamics, however, despite the large

number of highly successful applications, current phylogeny-based methods face a number of challenges. Phylodynamic tools and algorithms include highly computationally intensive stages, such as multiple sequence alignments and Bayesian Markov Chain Monte Carlo tree inferences, which often make them unable to scale to arbitrarily large data sets of samples. Phylogeny-based approaches in molecular epidemiology have room for improvement in allowing reliable inference of transmission history within transmission clusters, which is a crucial step in studying recent and emerging outbreaks. In Chapter III we outline our Boolean logic-based lightweight clustering algorithm that outcompetes existing heuristics in analyzing viral outbreaks and we hope that our platform will be utilized by public health officials in studying viral disease dynamics and guiding public health interventions.

After achieving successful clustering of microbial genomic data (as seen above), I moved on to utilize supervised clustering algorithms to analyze microarray and RNA-seq data for human cancer patients. Precision medicine for oncology is rapidly growing and evolving, and machine learning will very likely play a key role within clinical oncology moving forward. A key goal of precision medicine for cancer/oncology is the accurate prediction of optimal drug therapies based on individualized molecular profiles of patient tumors. For this project we gathered raw gene expression data from cell lines that had already been tested against a diverse set of small molecule libraries by the National Cancer Institute (NCI-60). I started to build drug specific machine learning models using this microarray data with accompanying drug response data (in the form of growth inhibition 50 or GI-50). I tested progressively better version controlled machine learning

algorithms on cancer cell lines including those from the recent NCI-DREAM Challenge. The highly accurate prediction results on cell lines encouraged me to apply this model to ovarian cancer patients from Northside Hospital (Atlanta, GA) and gathered by Dr. John McDonald's lab at Georgia Tech and several Gene Expression Omnibus (GEO) data sets from the National Center for Biotechnology Information (NCBI). Chapter IV discusses the details of this work which has now been published in PLOS ONE. I then continued to expand the program to analyze RNA-seq data and employed matched sets of RNA-seq data and drug-response profiles from The Cancer Genome Atlas (TCGA) database. By clustering patients into two groups (i.e. responders versus non-responders), we were able to predict optimal drug therapies for the individual patients. Chapter V discusses details of this work (which has now been submitted for publication).

CHAPTER 1. INTRODUCTION

With the advent of the era of microarray technology followed by next generation sequencing (NGS) platforms there is great promise that these technologies can revolutionize our approach to disease prevention, and cancer detection and treatment [1-3]. In 2005, we witnessed the beginning of the NGS era with platforms from companies such as ABI (SOLID), Roche (454) and Illumina (Genome Analyzer)[4]. These technologies have developed rapidly since this time and the cost of sequencing (per million reads) continues to decrease to present day. As data accumulates in public repositories we face the need to optimize algorithms to be able to scale well with the exponential increase in data [5]. Our solution is to utilize k-mer based alignment-free algorithms and heuristics that do not require canonical string matching/hamming distance calculations. k-mers typically refer to substrings of length *k* extract from a string. In sequencing analysis, k-mer refers to substrings of length *k* abstract from a string. Here, we used a sliding-window approach to obtain all k-mers and their frequencies. Microarray probes are composed of k-mers that are short sections of genes or other DNA elements, that are used to measure the expression level of large numbers of genes simultaneously or to genotype large numbers of genetic variants within a genome. Affymetrix microarray gene chips use k-mer probes of 25-mer oligos and output the expression level of those probe sets [6-9]. Our approach utilizes two main approaches: **a)** k-mer based un-supervised machine learning (ML) clustering algorithm that computes frequency counts of k-mer matrices and **b)** supervised ML clustering model on the

matrices of microarray probe data. The latter supervised ML cluster algorithm was specifically trained to predict optimal drug therapy for cancer patients. In the summary section above we mention how recent efforts using ML for precision medicine have benefited from community assessments of predicted drug response [10, 11]. Our core principle for the work presented in this dissertation revolves around building truly open source platforms (Fig. 1). From this work we have built version controlled software that utilizes publically available training and testing data which is able to **a)** interrogate 100k+ samples, **b)** be applied to many different type of data, **c)** optimize normalization to limit batch effects, **d)** reduce over fitting and eliminate substitution bias [12, 13] and finally **e)** accurately predict the ground truth state. By providing true open access to our software, we hope to encourage additional improvements to our methods with constructive comparisons with alternative approaches leading to the development of optimal strategies for personalized infectious disease and cancer medicine.

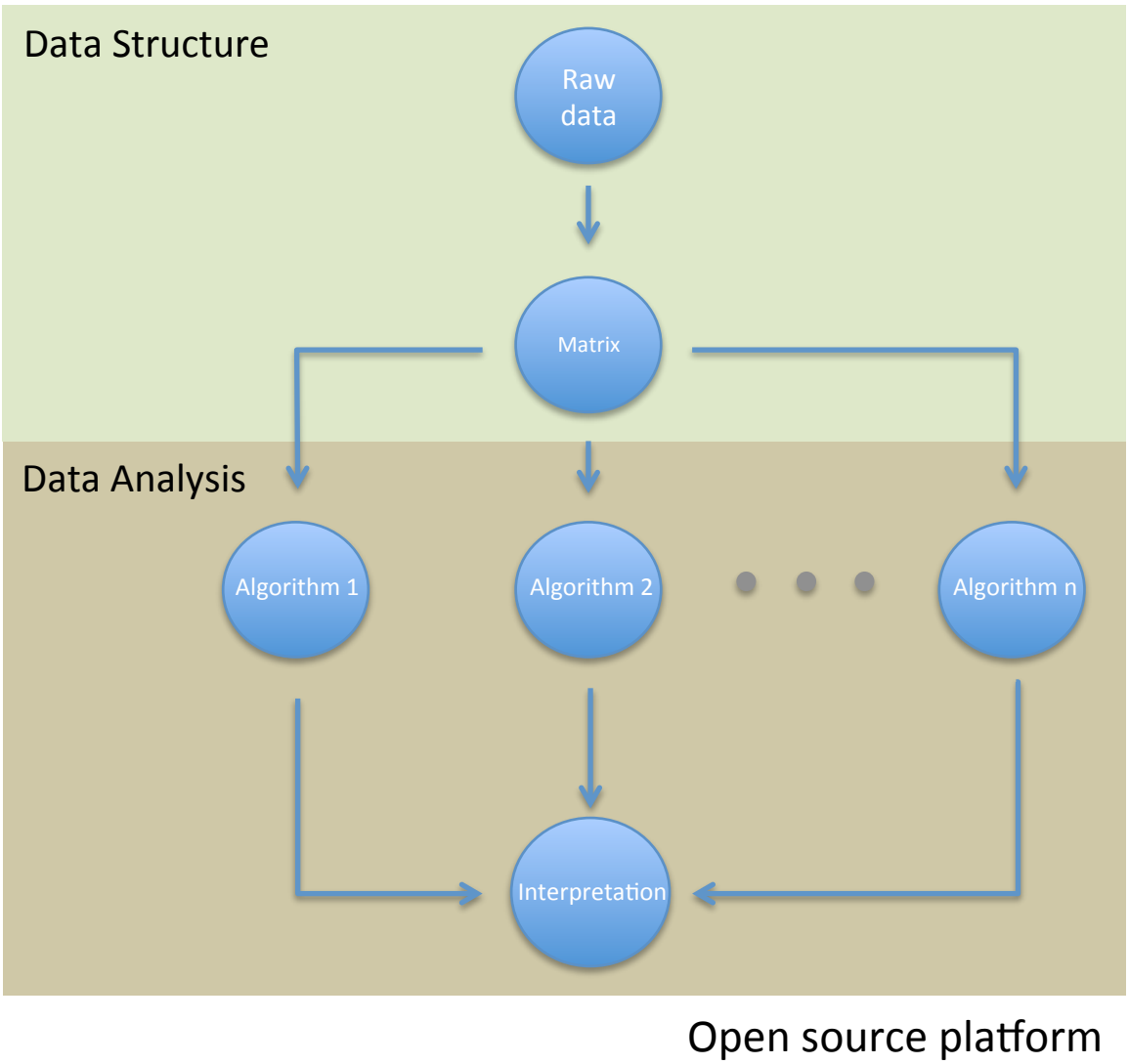


Figure 1 – Open source platform, which provides both Data Structure and Data Analysis.

1.1 Genomic Clustering

Genomic clustering is an important problem in bioinformatics [14, 15]. Significant progress with DNA-string matching algorithms continues. Among the well-known techniques of DNA-string matching are the Smith-Waterman algorithm [16] for local alignment, the Needleman-Wunsch algorithm [17] for global alignment, and the Hidden

Markov's model for multiple sequence alignment [18]. Although their contributions have been crucial to modern computational biology they do have their limitations. These methods all depend on a gap penalty alignment to a given reference and are also computationally intensive. Standard alignment and matching algorithms are known to be processor intensive even for the comparison of moderate length DNA sequences. Rapid advances in automated DNA sequencing technology has created a need for statistical summarization of large volumes of sequence data with the end goal of efficient and effective statistical analysis. There is a shortage of rapid and parsimonious procedures algorithms in this space. In Chapter II, we present our solution to this problem with utilizing an open source software package that utilizes an elegant and efficient alignment free k-mer approach [19]. By avoiding alignment, we can directly apply clustering algorithms on the numerical matrices derived from the DNA sequences. These clustering results are discussed in Chapter III. This approach presents an open source platform for genomic clustering based on a novel calculation for the pairwise distances between genomes.

1.2 Boolean analysis

Boolean analysis has been widely used in data mining and popularized by Flament (1976) [20]. By utilizing Boolean logical formulae, this approach can detect deterministic dependencies between the matrices of data that share homology. Some basic operations are conjunction $x \wedge y$ (AND), disjunction $x \vee y$ (OR), negation $\neg x$ (NOT), and exclusive-OR $x \oplus y$ (XOR). We term our approach Boolean when we formulate matrices of data into $n>=1$ to 1 and $n=0$ to 0 (i.e. algebra with two values). In this rubric 1 is taken to be

represent true (**T**) and 0 is taken to represent false (**F**). In Chapter II, we have defined the pairwise distance of two genomic sequences based on the exclusive-OR operation. Each sequence is fragmented into small k-mers, and the presence of a k-mer is considered as 1, the absence as 0. The Boolean analysis of two sequences then becomes an algebraic function of these two binary values as mentioned above.

1.3 Machine learning on bioinformatics

Machine learning is the development of algorithms and techniques that allow a computer to learn. It has a broad spectrum of applications such as stock market analysis, cheminformatics, and bioinformatics to name a few. Recently, the amount of biological data requiring analysis has exploded, and many studies have been using machine learning methods to analyze this data [21, 22]. Hence, machine learning in bioinformatics has become an important research area for both computer scientists and biologists. It requires the development of tools and methods capable of transforming all of this heterogeneous data into biological knowledge about the underlying mechanism. These tools and techniques should allow us to go beyond a mere description of the data and provide knowledge in the form of testable models. With this work we will be able to create simplified abstractions that constitute a model heuristic and from this we will be able to obtain predictions from this system. There are several biological domains where machine learning techniques are applied for knowledge extraction from data.

In a modeling problem, the 'learning' term refers to running a computer program to create a model by using training data or experience. Machine learning uses statistical

theory when building computational models since the objective is to make inferences from a sample. The two main steps in this process are to induce the model by processing huge amounts of data and to represent the model and make inferences efficiently. It must be noted that the efficiency of the learning and inference algorithms, as well as their space and time complexity, transparency and interpretability, can be as important as their predictive accuracy. The process of transforming data into knowledge is both iterative and interactive. The iterative phase consists of several steps. In the first step, we need to integrate and merge the different sources of information into one single format. The detection and resolution of outliers and inconsistencies is an important first step and we employ various techniques to solve this including data warehouse techniques. In the second step, it is necessary to select, clean and transform the data. As shown in Chapter II, we utilize a *pairwise normalization* approach to ensure that the microarray data for each patient is on the same scale as the learning set and the other patient samples. This is then followed by the next step which selects relevant and non-redundant variables. This also can be referred to as *feature selection algorithms* we used in Chapter IV, and the most effective algorithm on our data set, *recursive feature elimination* (RFE). In the fourth step, called *data mining*, we take the objectives of the study into account in order to choose the most appropriate analysis objective for the data being analyzed. Once the version controlled model is obtained, it should be evaluated and interpreted both from statistical and biological points of view and, if necessary, we should return to previous steps for a new iteration to improve the model and then increment the version control number. This includes the resolution of conflicts using the current knowledge in the

domain. We discuss the optimization of our program in Chapter VI. Optimization problems can be regarded as the task of finding an optimal solution in the space of many possible solutions. The choice of the optimization method is crucial for the solution of the problem. Optimization approaches to biological problems can be classified into exact and approximate methods. Exact methods output the optimal solutions when convergence is achieved. However, they do not necessarily converge in every instance. Approximate algorithms always output a candidate solution, but it is not guaranteed to be the optimal one. Optimization is also a fundamental task when modeling from data. In fact, the process of learning from data can be regarded as searching for the model that gives the data the best fitting. In this search, in the space of all possible models, any type of heuristic can be used. Therefore, optimization methods can also be an ingredient in modeling.

Bioinformatics is the development and application of computational methods for management, analysis, interpretation, and prediction, as well as for the design of experiments. The goal in machine learning is to extract useful information from a body of data by building good probabilistic models and to automate the process as much as possible.

1.4 Precision medicine in cancer treatment

Precision medicine is an approach to patient care that allows doctors to select treatments that are most likely to help patients based on a genetic understanding of their disease [23, 24]. This approach, also called personalized medicine, is not new, but recent

advances in science and technology have helped speed up the pace of this area of research.

In present day when you are diagnosed with cancer you usually receive the same treatment as others who have the same type and stage of cancer. Even so, different people may respond differently and until recently doctors didn't have a way of formally understanding why. After decades of research, scientists now understand that patients' tumors have genetic changes that cause cancer to grow and spread and these are being elucidated in greater numbers and greater detail over the past decade. Researchers have also learned that the changes that occur in one person's cancer may not occur in others who have the same type of cancer. And, the same cancer-causing changes may be found in different types of cancer.

Optimal prediction of personalized cancer drug therapies using machine learning is an area of intense research activity in recent years and numerous community assessments have been carried out in pursuit of maximal sensitivity and specificity [25-28]. As mentioned we were frustrated at how little code was actually shared publically by this previous work and we sought to change that trend by creating a framework for truly open source, version controlled software that utilizes publically available training and testing data, using a highly versatile support vector machine (SVM) algorithm utilizing standard recursive feature elimination (RFE) methods to predict cancer drug response.

1.5 Chapters

CHAPTER II outlines the methods used on k-mer generation, clustering and modeling. This chapter gives the definition of k-mer and dataset we used on learning and testing.

CHAPTER III discuss the genomic clustering based on k-mer alignment free approach. This chapter shows the clustering result on several data sets, and preparing the clustering algorithm for modeling and predicting on CHAPTER IV

CHAPTER IV presents a novel machine learning algorithm that predicting optimal cancer therapy.

CHAPTER V discusses a novel application of the predicting model from CHAPTER IV.

CHAPTER VI concludes this dissertation and feature steps for my projects.

CHAPTER 2. MATERIALS AND METHODS

2.1 k-mer

2.1.1 Definition

Nucleotides A, C, G, and T are assigned as [00, 01, 10, 11] and then we used a binary data structure to maintain them in memory. For example, a sequence ATTCG would be transferred to [00, 11, 11, 01, 10], which can be stored as a decimal number, 246. In sparse matrix form we array these into a vector sequence based upon the magnitude of k-mer (Fig. 2A), and most of our analyses are based on the vector format. We demonstrate the conversion into matrix format of k-mer data in Figure 2B. Once the k-size was determined we retain the count of each k-mer.

We specify that the matrix is *n x m* with the definition that $n, m \in \mathbb{Z}^*$

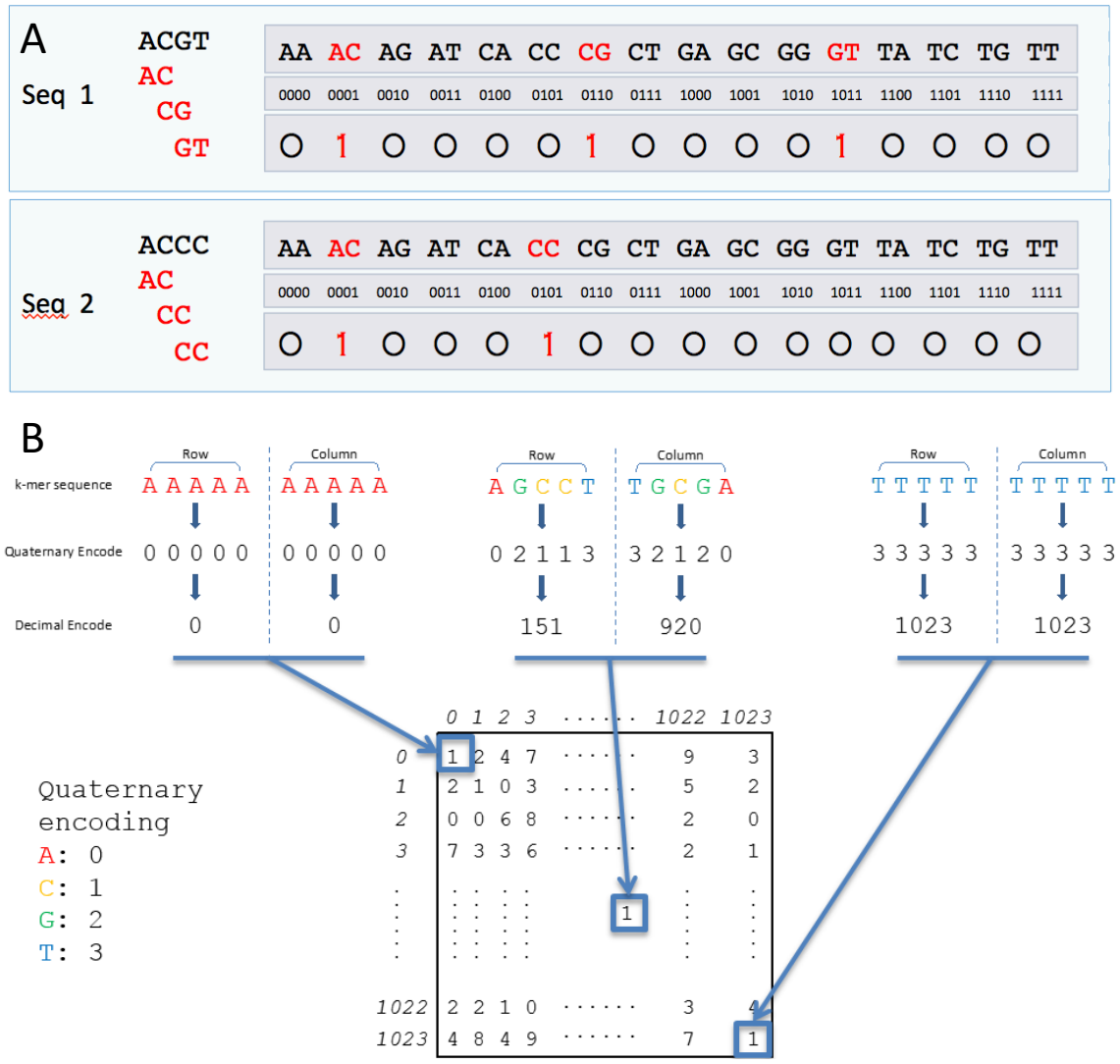


Figure 2 – Universal k-mer encoding. Universal encoding defined using the Quaternary-Decimal encoding system. A) vector form B) matrix form.

In our k-mer encoding, there is an equivalency between our matrix (τ) and its linear transformation (τ), which can be seen as interchangeable.

$$\begin{bmatrix} f_{11} & f_{12} & \cdots & f_{1m} \\ f_{21} & f_{22} & \cdots & f_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ f_{n1} & f_{n2} & \cdots & f_{nm} \end{bmatrix} \underline{\text{Linear transformation}} \begin{bmatrix} f_{11} & f_{12} & \cdots & f_{1m} & f_{21} & f_{22} & \cdots & f_{2m} & \cdots & f_{n1} & f_{n2} & \cdots & f_{nm} \end{bmatrix}$$

Where f_{ij} is frequency of k-mer in the sequence. We present both format for different application of analysis.

2.1.2 Investigating relationship of k-mer occupancy (τ) and Shannon entropy to optimize k-mer length

Shannon entropy is the average unpredictability in a random variable, which is equivalent to its information content, defined by equation number (2).

$$H(\mathrm{s}) = -\sum_{i=1}^{N} p(\mathrm{n}_i) \log_b p(\mathrm{n}_i) \tag{2}$$

Where $H(\mathrm{s})$ is the Shannon entropy of the genome sequence (s), N is the possible outcomes [A, C, G, T], $p(\mathrm{n}_i)$ is the probability of occurrence of each nucleotide in that sequence.

k-mer occupancy τ is the calculated as the proportion of k-mers divided by the total number of possible k-mers given by equation number (3).

$$\tau(s,k) = \frac{\sum_{i=1}^{N^k} m_i}{N^k} \quad m \in \{0,1\} \tag{3}$$

where $\tau(s,k)$ is the k-mer occupancy of the genome sequence (s), N is the possible outcomes [A, C, G, T], m_i is the occurrence of each possible k-mer in that sequence. Taking in Shannon entropy and k-mer occupancy τ, we plot τ divided by the log of

genome size (L) on the y-axis and the H(s) to the power of 10 on the x-axis as defined below bacterial genomes as defined by the following

$$\text{y: } \frac{\tau(s,k)}{\log_{10}(L)}, \text{ x: } H(\text{s})^{10} \tag{4}$$

Optimizing k-mer length for a given set of k-matrices and equivalent linear vectors relies on local optimal slope (m) of the overall trend line of r^2 relationships of the dataset, assuming a variance of genome lengths less than 30% of the overall variance.

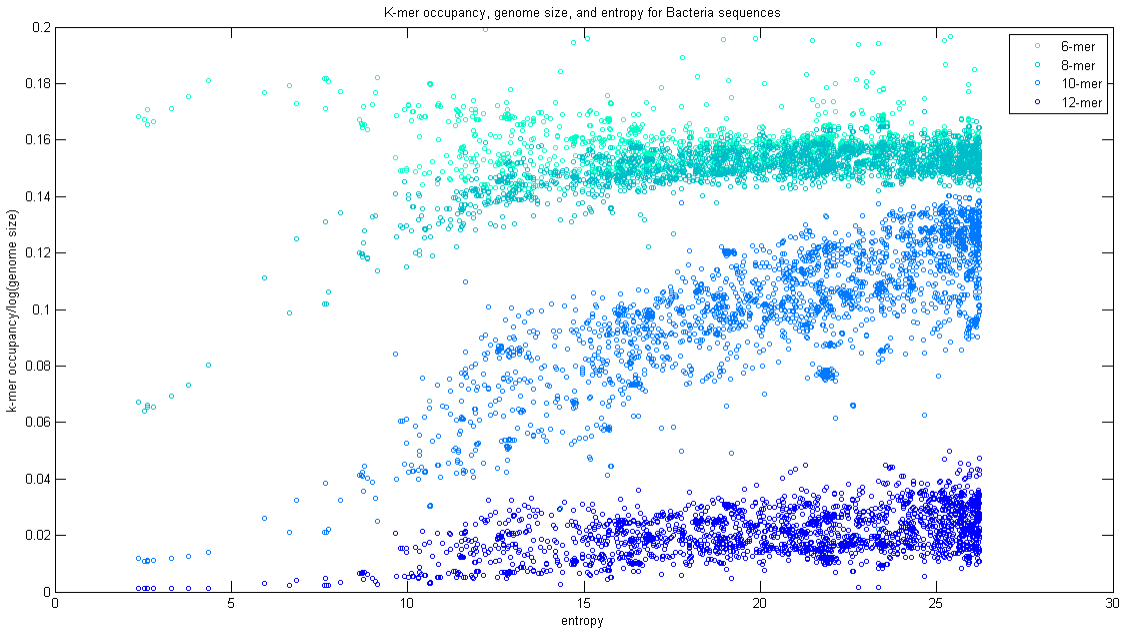


Figure 3 – Different k-mer lengths and k-mer occupancy, genome size, and entropy for 2569 bacterial genomes.

Given a linear polynomial function for these data points, we will attempt to maximize a within this function (i.e. slope) to allow for the optimal *k* size for a given data set with a limited amount of variance (specify here) in genome length (***I***)

$$\frac{\tau(s,k)}{\log_{10}(L)} = aH(s)^{10} + b \quad (5)$$

Fit linear polynomial function (Equation 5) to Figure 3, we can get that k=10 give us the optimal slope in equation number (5).

This relationship allows for determination of the ideal k-mer length to be utilized for a given subset of samples and confirms that genome encoding will be ideal by optimizing the slope of the equation number (5) to a maximum given a strong r^2 relationship of the data points with the desirable outcome of increasing size and complexity yield higher values, but the maximization of the slope for a set of k-mer lengths prevents overly sparse (Fig. 3, k=12) or overly saturated (Fig 3, k=6) matrix comparisons.

2.1.3 Discrete Mathematics on τ matrices Differential Occupancy

We then went on to carry out discrete mathematics on these compiled matrices to determine the extent to which we could define genome similarity using this encoding. One simple metric is the resultant value of the XOR Boolean operator between two sparse vectors, which we here define as delta tau ($\Delta\tau_k$), or differential occupancy, where k defines the k-mer length. We define the term differential occupancy as the XOR function between two sparse vectors that derives the resultant sparse vector followed by summing the number of non-zero elements then dividing by the total number of possible k-mers, which is 4^k (Equation 6).

$$\Delta\tau_k = \frac{sum(\tau_{seq1} \otimes \tau_{seq2})}{N^k} \quad (6)$$

We provide all data as to the pairwise differential $\Delta\tau_k$ encoding between species p and q specifying k-mer length. Although these pairwise $\Delta\tau_k$ calculations took ~48 hours of computation time on a standard Intel i5 processor with 32 GB of RAM. This Boolean encoding system allows for hardware accelerators such as field-programmable gate array (FPGA) processors to quickly compare this data with specific combinations of half and full adder circuits. Indeed, this Georgia Tech encoding in part is envisioned in an ecosystem where memory and compute capabilities make it possible for such large-scale comparisons and allows for complete sparse matrix calculations over increasing k-mer size. Using this $\Delta\tau_k$ we carry out hierarchical clustering and obtain expected clustering of these bacterial genomes. It should be noted that this analysis is not a phylogenetic analysis, rather analysing global genomic content, but will detect shared lineage as well as putative horizontal gene transfer events.

The pseudo code for calculating Differential Occupancy is as follows:

Algorithm 1 Differential Occupancy

```
inputdata ← fasta or fastq sequnces;
procedure KMERIZE
    Ksize ← k;
    ls *fasta — xargs -I one java -jar kanalyze.jar count -k Ksize
    -o one-Ksize.kc -f fasta -p kanalyze.outfmt=int one;
end procedure
procedure DIFFERENTIAL OCCUPANCY
    kfp ← path to kc files;
    R -q -e "library(Finch);DF=Finch.dif(Ksize,'.');write.csv(DF, file = 'DF.csv')";
end procedure
```

2.2 Matrix format

We present a matrix format to store transformed sequences, which we term k-matrix. Each column represents one sample, and k-mer frequencies are stored in each row. This format benefits on data aggregation and analysis. For example, we used matrix format for denoising in Chapter III. We sum the appearance of each k-mer in the last column to carry out further processing.

$$\begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{nm} & a_{n2} & \cdots & a_{nm} \end{bmatrix} \begin{bmatrix} Sum_1 \\ Sum_2 \\ \vdots \\ Sum_n \end{bmatrix}$$

Where *m* is the number of samples, and *n* is equal to 4^k, all possible number of k-mers. a_{ij} is the appearance of k-mer, which can be 0 or 1. The sum of each k-mer appearance among *m* samples are saved as sum_i. Then we remove k-mers (rows) which sum_i is smaller than certain threshold. In Chapter III, we dynamically calculate threshold based on z-score of 2.

2.3 Affymetrix microarray

Microarray data normalization. Patients' gene expression microarray (.CEL) files are analyzed by using the Affymetrix Microarray Analysis tool, expression console (http://biology.usf.edu/cmmb/research/GeneAtlas%20Info/expression_console_userguide.pdf). Both the library and annotation file for the microarray data we analyzed are HG

U133 Plus 2.0. The MAS5 normalization is chosen for this analysis. Export probe set result and annotations as plain text file.

Drug response prediction. Binary classification is performed for the patient's drug response prediction. The prediction score for test samples are calculated by using the decision function as follows:

$$prediction\,score = -1 \times (\sum_{f=1}^{i} w_f x_f + b) \quad (7)$$

Where w and b are the weight vector and bias parameters from the DrugResponse R packagee. The input x is the normalized patient sample gene expression data with RFE selected i number of features. The classification of the patient drug response is based on this score. We call the sample a 'responder' to the drug if this score is higher than 0, and a 'non-responder' to the drug if the score is lower than 0.

2.4 TCGA RNA-seq

We downloaded TCGA RNA-seq data, normalized gene expression quantification processed following the NCI mRNA analysis pipeline (https://docs.gdc.cancer.gov/Data/Bioinformatics_Pipelines/Expression_mRNA_Pipeline/). Then we did data cleaning. Genes are removed if more than 25% of the samples have zero expression value. Then we download the drug response labels for all samples. Label 1 represents responder, and 0 represents non-responder. Once complete, we randomly select 75% from responders and 75% from non-responders into training data set. The rest

25% samples are testing data set. The data set contains two drugs, Gemcitabine and Fluorouracil and these two are the only ones with sufficient patients to train the machine learning algorithm. Within this sample set there are 92 patients profile for Gemcitabine, and 60 patients profile for Fluorouracil. NCI classifies patients into four stages: 'complete response', 'partial response', 'progressive disease', and 'stable disease' (https://www.cancer.gov/publications/dictionaries/cancer-terms). Complete response is defined by the disappearance of all signs of cancer in response to treatment, although this does not always mean the cancer has been cured (i.e. 'complete remission'). Partial response indicates a decrease in the size of a tumor, and/or the extent of cancer in the body, in response to treatment (i.e. 'partial remission'). Progressive disease is defined as a cancer that is growing, spreading, or getting worse. Stable disease indicates that the cancer is neither decreasing nor increasing in extent or severity. For this thesis we simplify these into two classes: 'responders' which includes both 'complete response' and 'partial response'; and 'non-responders' which includes 'progressive disease' and 'stable disease'. This data set is randomly separated into a 75% training set and a 25% testing set. A SVM models is built using the learning set data, and evaluated using the test data set. Recursive feature elimination (RFE) was performed to find the minimum set of features that maximized accuracy in the classification on the test dataset. The approach starts by removing the least relevant 100 features for the model from the bottom (lowest weights) of the sorted feature list. The following SVM model is built using the remaining features, and then again removes the 100 features with lowest weights. This process precedes recursively until the number of remaining features reaches 100. Thereafter,

features are removed one at a time until the most informative set of features is obtained. After the RFE, the model will pick the most informative set of features yields the highest accuracy testing on test set. The predictive model for each drug is based on the most informative set of features determined for that drug. Leave one out cross-validation (LOOCV) is used to evaluate the performance of each of the models as previously described.

2.5 Drug response data

2.5.1 TCGA data

Clinical trial data (12,051 records) were downloaded from TCGA for 32 types of cancer. The data was cleaned as described in the pre-processing methods, and 2 models were selected encompassing 140 patients. Corresponding Upper-Quartile Normalized Fragments Per Kilobase per Million mapped reads (UQ-FPKMs) from patient primary tumor samples were downloaded using the Genomic Data Common's API.

The clinical data needed to be assessed to determine which were the best drugs to investigate. We first defined the definition of responder as a patient who had partial or complete response and a non-responder as clinical progressive disease or stable disease. Next, to determine which drugs were viable candidates for the analysis, we required a sample size of at least 30 patients for the drug of interest with at least 15 for each type of response. The clinical trial data were then cleaned using fuzzy matching and manual curation to ensure consistency of drug name and formatting, and only patients on a single candidate drug at a time were retained for evaluation. From this list, the candidate drugs

that had the highest number of patients were chosen with balance and availability of RNAseq data being kept in mind as a constraint.

2.5.2 Ovarian cancer data

Samples of primary tumors collected from 23 ovarian cancer patients were snap frozen in liquid nitrogen within one minute of surgical removal and transferred to the lab for laser capture microdissection of cancer cells and subsequent microarray gene-expression analysis (Affymetrix, U133Plus 2.0 arrays). Informed patient consents were obtained under appropriate Georgia Institute of Technology Institutional Review Board protocols (H14337).

Patient responses to administered chemotherapies were monitored by measurement of CA-125 values prior and subsequent to treatment according to standard criteria [13]. Patients were considered to be responsive to treatments if their respective CA-125 values dropped and remain below normal values (<35) within 60 days of the start of chemotherapeutic treatment.

Microarray gene expression and patient drug response data were uploaded to our predictive algorithms and predictions were generated as described in 2.6.2.

2.6 Cluster algorithms

2.6.1 Hierarchical Principal Component Analysis (hPCA)

Hierarchical Cluster Analysis (HCA) is a clustering method which explore the organization of samples in groups and among groups depicting a hierarchy [29]. We applied HCA on principal components of our data set, so called hierarchical Principal Component Analysis (hPCA). The principal components are calculated by using an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables. This transformation is defined in such a way that the first principal component has the largest possible variance, and each succeeding component in turn has the highest variance possible under the constraint that it is orthogonal to the preceding components. In this project, we used first 3 principle components. The first 3 principle components contain 86% of all information, and we only gain incremental extra information if we continue to add components (i.e. adding 16 additional components adds only 4% additional information). Then we utilized Euclidean distance function to calculate pairwise distance between samples, and construct clustering based on the pairwise distance. The result of hPCA is presented in a dendrogram, a plot which shows the organization of samples and its relationships in tree form.

2.6.2 K-means clustring

K-means clustering is a method of cluster analysis that aims to partition n observations into *k* clusters in which each observation belongs to the cluster with the nearest mean. We choose the string kernel:

$$k(x,x') = <\phi(x),\phi(x')> H \tag{10}$$

In equation number (10) the term *x* is the set of discrete structures (e.g. the set of all documents), and *H* is a Reproducing Kernel Hilbert Space (RKHS). In our k-mer data we can describe the frequency of k-mer appearances in the genome using the following equation:

$$k(x,x') = \sum_{s \subseteq x, s' \subseteq x'} w_s \delta_{s,s'} = \sum_{s \in A^*} w_s num_s(x) num_s(x') \tag{11}$$

That is, we count the number of occurrences of every k-mer s in both *x* and *x'* and weigh it by *Ws*. We only consider the 8-mer, so we assign other k-mer's weights to 0. For K-means, we randomly chose *k* central points and calculate the mean of the k-mer vectors to represent the center points. Then sign each data point to its nearest central point by the distance equation number (12), and update the central points. After several iterations, we can find the best clustering after iterations when the central points stop changing.

$$D(X,X') = \| X - X' \|^2 \tag{12}$$

2.6.3 SVM

Support Vector Machine (SVM) is a discriminative classifier widely used in clustering problems. A support vector machine constructs a hyperplane or set of hyperplanes in a high-dimensional space, which can be used for classification. A good separation is achieved by the hyperplane that has the largest distance to the nearest training-data point (vectors) of any class. We use SVM algorithm to build a model on

training data and assign new examples to one category or the other. Models are built using learning set data, and evaluated using test data set. Linear support vector machine (SVM) is employed recursively as a classification model to separate samples into two classes: sensitive and resistant. The learning function is *svmtrain* (Matlab R2013b version 8.2.0.701), and the kernel function is linear. The samples are represented as a vector $\boldsymbol{x}$, and the two classes are divided in the dataspace by a hyperplane $\boldsymbol{wx' + b = 0}$ that maximizes the margins between the learning samples of the two classes. This margin is defined such that:

$$\begin{aligned} wx'+b \geq 1,\ c=1 \\ wx'+b \leq -1, c=-1 \end{aligned} \tag{8}$$

Binary classification is performed for the test prediction. The prediction score for test samples are calculated by using the decision function as follows:

$$prediction\,score = -1 \times (\sum_{f=1}^{i} w_f x_f + b) \tag{9}$$

Where w and b are the weight vector and bias parameters from the SVM model. The input x is the normalized test sample gene expression data with RFE selected i number of features. The classification of the patient drug response is based on this score. We call a sample responder to the drug if this score is higher than 0, and non-responder to the drug if the score is lower than 0.

2.7 Feature selection

2.7.1 Principal Component Analysis (PCA)

We utilized Principal Component Analysis (PCA) to obtain feature selection by use the most important components of our data sets. The principal components are calculated by using an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables. In this project, we used first three principle components.

2.7.2 Sequential forward selection

Sequential forward selection starting from an empty feature set, and creates candidate feature subsets by sequentially adding each of the features not yet selected. For each candidate feature subset, it performs 10-fold cross-validation by repeatedly calling fun with different training subsets of training data until there is no improvement in prediction. From this we choose a subset of features for the final analysis.

2.7.3 Recursive feature selection

Recursive feature elimination (RFE) was performed to find the minimum set of features that maximized accuracy in the classification on the test dataset. The approach starts by removing the least relevant 100 features for the model from the bottom (lowest weights) of the sorted feature list. The following SVM model is built using the remaining features, and then again removes the 100 features with lowest weights. This process proceeds recursively until the number of remaining features reaches 100. Thereafter,

features are removed one at a time until the most informative set of features is obtained [30-32]. If there are multiple models with the highest accuracy, the model with the fewest number of features is selected. Each model is forced to contain a minimum of 10 probes. The predictive model for each drug is based on the most informative set of features determined for that drug. Leave one out cross-validation (LOOCV) is used to evaluate the performance of each of the models.

The pseudo code for the RFE is as follows:

Algorithm 2 SVM: RFE

```
inputdata ← TCGA RNAseq normalized expression value;
remove missing data;
procedure SORT FEATURES
    svm ← svmtrain on inputdata(allFeatures);
    remainFeatures ← allFeatures;
    array sortedFeatures[];
    n ← 100;
    while remainFeatures > 0 do
        if remainFeatures < 100 then
            n ← 1;
        end if
        svm ← svmtrain on inputdata(remainFeatures);
        w ← svm weights;
        sort remainFeatures base on w;
        remainFeatures ← remainFeatures - last n features;
        sortedFeatures ← [last n features,sortedFeatures];
    end while
end procedure
procedure SVM TEST
    array acc[]
    for i 1:length(sortedFeatures) do
        svm ← svmtrain on inputdata(sortedFeatures(1:i));
        classOut ← svmclassify on test data;
        acc(i) ← CorrectRate of classOut;
    end for
    bestModel ← sortedFeatures(max(acc));
end procedure
return bestModel
```

2.8 Normalization

Individual gene expression microarray (.CEL) files are normalized one by one against the original NCI 60 gene expression microarray data specific to each array (both Affy U133 Plus 2 and Human Exon Array) using standard quantile normalization [33, 34] and using the mean of each probe. This approach creates distributions for each array that are as similar as possible in terms of statistical properties.

CHAPTER 3. LINEAR ALGEBRAIC AND BOOLEAN ANALYSIS OF GENOMIC SEQUENCES

3.1 Abstract

To date the comparison of genomic sequences [35] have routinely utilized shorter conserved regions for comparative genomics. Genomic analysis has become one of the major tools for disease outbreak investigations. Current phylogenetic analysis [36] therefore can create divergent results based on which genetic loci are utilized for this analysis. Sequence similarity is also commonly determined by first carrying out gap penalty pairwise alignments [37-41] for a set of sequences, and the similarity is quantified based upon this alignment. This approach, however, has limitations and is then primarily utilized to compare relatively conserved sequences and is also dependent upon the algorithm utilized to create the alignment. This impedes genomic analysis of outbreaks of highly mutable viruses associated with chronic infections, such as human immunodeficiency virus (HIV) and hepatitis C virus (HCV). From first principles we here develop a framework for encoding nucleic acid sequences into fixed length sections (k-mers) [42, 43] which we utilize to create an invariant pairwise distance of total information content of over ~4,000 bacteria [44] and viruses, revealing important cryptic relationships not previously reported. Also, our experimental validation shows that they can successfully identify genetic relatedness between viral populations, as well as clusters.

3.2 Introduction

k-mer strategies have been utilized by nearly all short read DNA alignment strategies, including de Bruijn-type [45, 46] combinatorial mathematics and Burrows-Wheeler transform (BWT) [47] algorithm to efficiently align sequences. Here we present our software, Finch, which adds another layer of abstraction to this field by creating an invariant encoding heuristic for k-mers that allows for the efficient analysis and computation of genome similarities (https://github.com/chuang95/Finch). This invariant encoding framework is computable by linear algebra and also diverse Boolean, logic and bit operations of discrete mathematics. In addition to orthogonal transformation of this dataset we also show pairwise differential occupancy (which we term $\Delta\tau_k$) of these data structures for hierarchical clustering of genomic sequences. To our knowledge we define the first global clustering of entire bacterial and viral genomes, which we define here as genomic topology, which led to the discovery of a number of novel cryptic genomic associations between species. Finch allows for this formalization of genomic relationships and yields an invariant pairwise differential occupancy ($\Delta\tau_k$) metric between all species based upon global information content of each genome, irrespective of Kingdom, and finally allows a universal complete encoding for species that was lacking in previous approaches.

The exponential growth in sequencing information of organisms of all Kingdoms has greatly increased our understanding of the diversity of genomes and genomic complexity. In order to help formalize and explore genomic encoding we have created a

sparse k-mer encoding system wherein genomic data is stored in an identical format that allows for global computations across all genomic data, regardless of homology. There has been a distinct lack of fundamental organization of genomic data, with primarily ad hoc analysis of genome similarity using subsets of evolutionarily conserved genes. We created a universal encoding system to store k-mer data in sparse linear vector form as well as an equivalent *n x n* square sparse matrix, which defined in Chapter II. Briefly, our system utilizes a sparse vector that contains 4k elements, arrayed in ascending order by the binary representation of the k-mer, where [A=00, C=01, G=10, T=11]. For these sparse vectors, we define tau (τ) by the number of non-zero elements divided by 4^k elements, where *k* represents the k-mer length. We routinely also transform this binary k-mer representation into a decimal number as seen in Figure 4. In order to optimize this k-mer analysis we authored KAnalyze [19], which is a fast and extensible k-mer suite with APIs specific for this process.

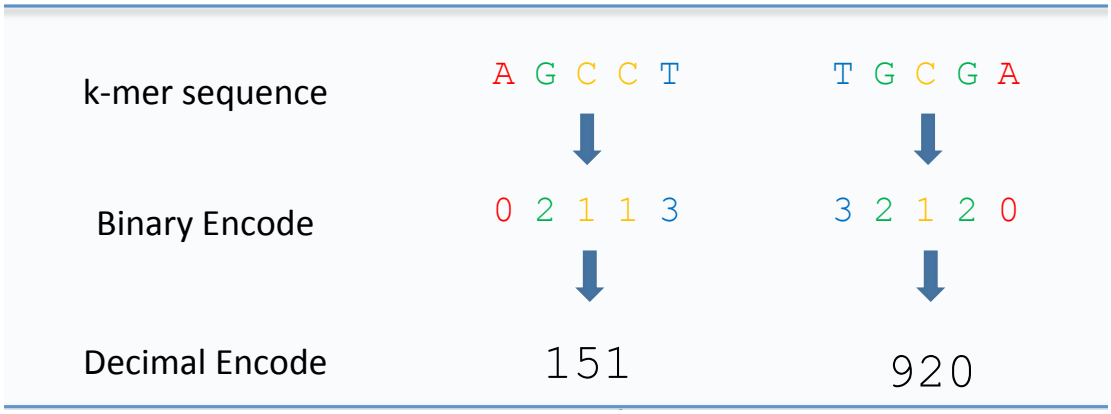


Figure 4 – Universal encoding defined using the Quaternary-Decimal encoding system.

3.3 Result

3.3.1 Kmeans clustering on viral sequence

We use a data set of 3 group of 8 virus: 2 different Dengue viruses, HIV-1, HIV-2, HHV2, HHV1, HHV4, and HHV4.1. Before implement the algorithm, we plot the distance matrix as heat map (Fig. 5) to show the pattern of the 8 viruses.

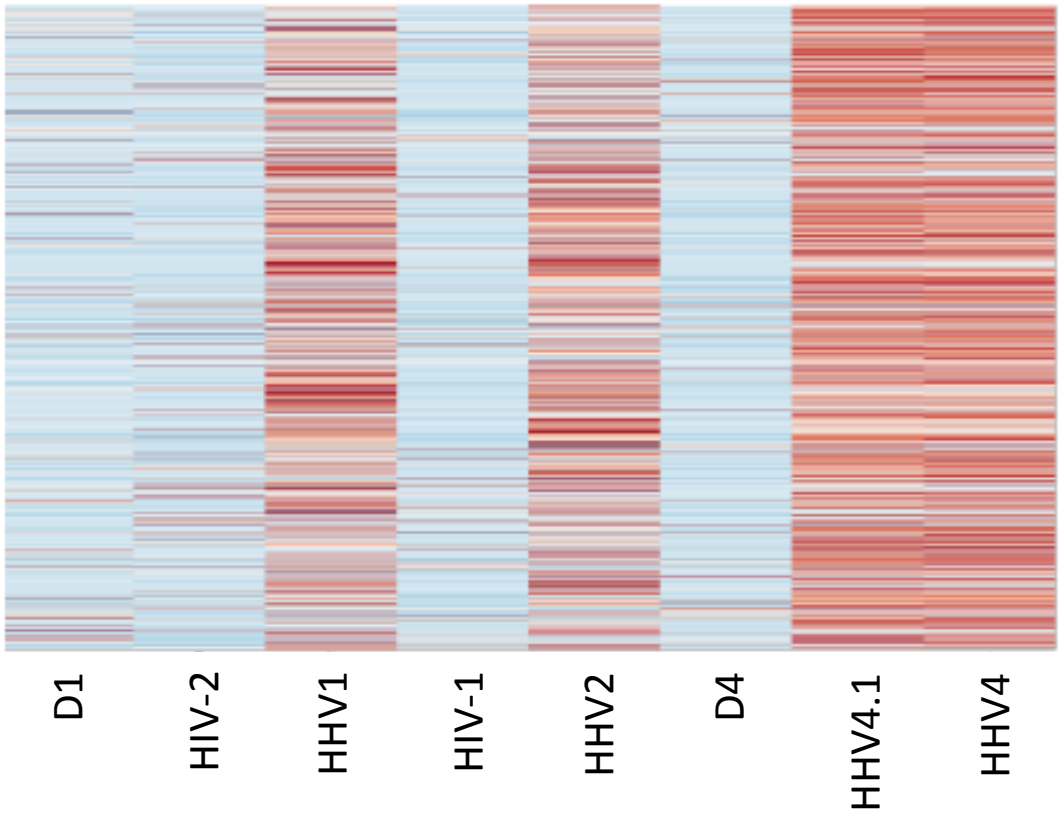


Figure 5 – Heat map of 8-mer frequency of 8 viruses. X axis are viruses and Y axis are 8-mer frequency.

We chose 3 center points, and the cluster result shown in Table 1 is concordant with the heat map shown in Figure 5. Base on this result, we applied the kernel kmeans to all viruses in our database, and choose *k* equal to 100. For later test, we will compare the

test data point with the 100 center point. It reduces the comparing complexity then comparing the test data with the whole database which is the strategy of Multiple Sequence Alignment (MSA).

Table 1 – Kernel K-means clustering.

Group	Virus			
0	D1	D4		
1	HIV2	HIV1		
2	HHV2	HHV1	HHV4	HHV4.1

3.3.2 Support vector machine (SVM) classification on viral and Bactria sequences

Support vector machines (SVM) are supervised learning models with associated learning algorithms that analyze data and recognize patterns, used for classification and regression analysis. In the real clinic sample, we need to separate the virus from the human gene and other possible spices. Our application takes an input as training data set and finds a boundary between the clusters. Then take the boundary points to define a support vector, and finds the maximum-margin hyperplane that divides the data points from the other ones belong to another cluster. Also, we used kernel trick to define the mapping from linear from the linear feature space to non-linear space that maximizes separation. The Gaussian *'rbf'* kernel function is chosen here.

First, we implemented the SVM algorithm in Matlab and apply it to virus sequence and random sequences. The training data includes the 10 previous viruses along

with 10 random sequences with different length and GC content. The test data includes 10 randomly chosen viruses (from NCBI) and 10 further random sequences. By using kernel function *'rbf'*, the accuracy rises to 80 percent (Fig. 6).

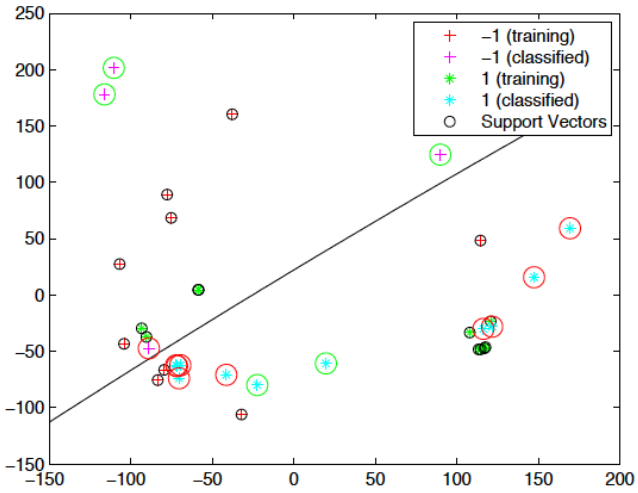


Figure 6 – SVM clustering (viruses in red, random sequences in green).

Furthermore, we applied SVM algorithm to separate viruses and bacteria sequence. Training data set is random chosen from our database, and test is with all 1756 viruses and 2271 bacteria sequence. The result is encouraging, and shows the pattern of the two groups. Also, we compare the SVM result of non-scaled (Fig. 7A, B) and scaled data (Fig. 7C, D). The accuracy (Table 2) of scaled data is higher because the different average sequence length separates the data well, however this measure does not help resolve true genomic complexity and the scaled versions should be utilized instead for actual analysis of sequences.

Table 2 – Accuracy measurements of non-scaled and scaled test.

Group	Virus error	Bacteria error	Accuracy
non-scaled	1	42	0.99
scaled	429	108	0.87

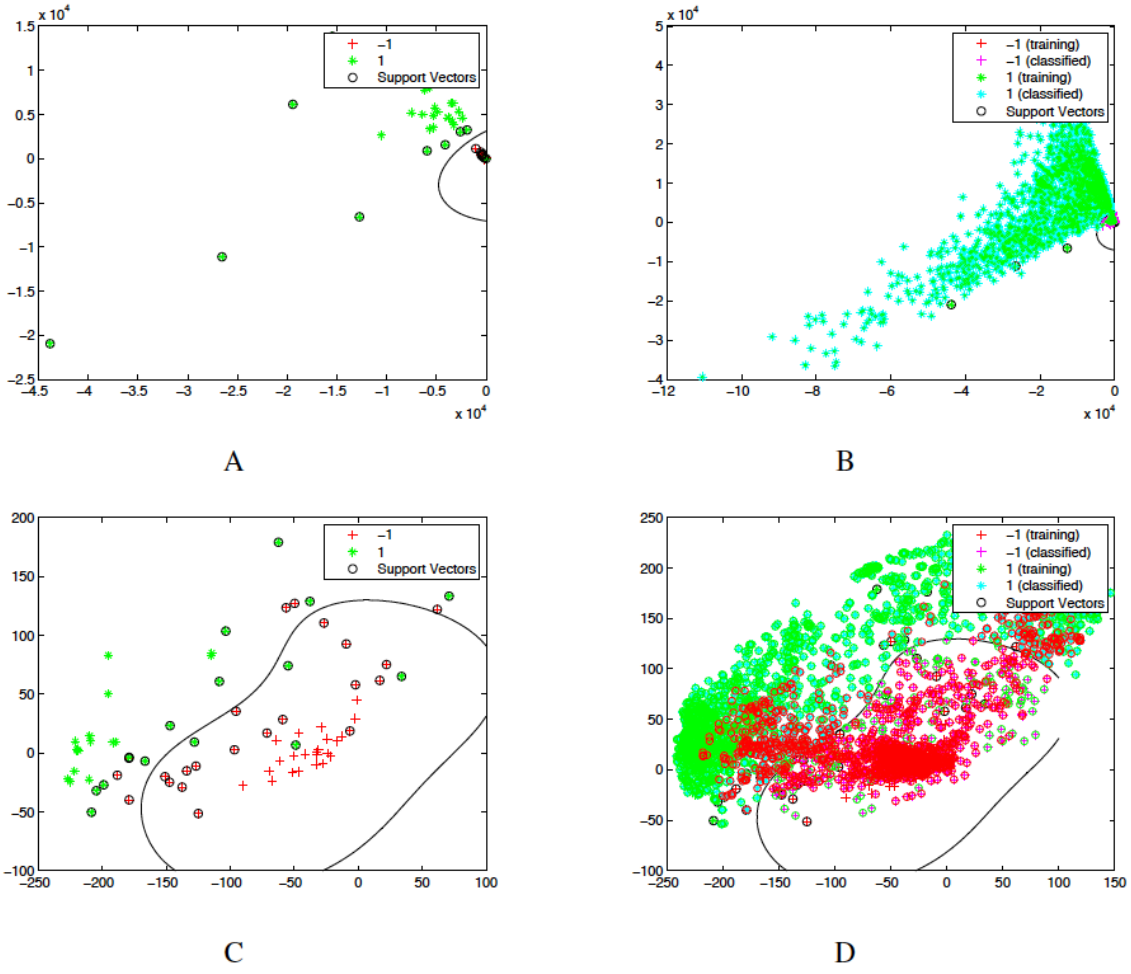


Figure 7 – Non-scaled data SVM training (A) containing the training set contains 35 viruses and 34 bacteria, and testing all 1756 virus and 2271 bacteria (B). Scaled data SVM training on 35 viruses and 34 bacteria (C) and SVM testing on all 1756 virus and 2271 bacteria (D). Viruses: red, bacteria: green.

3.3.3 PCA and Boolean analysis

From these first principles, we sought ways to globally analyze these sparse linear vectors. Using standard linear algebra approaches it is possible to complete orthogonal transformations of this data (Fig. 8A). In one manifestation we carry out hierarchical principle component analysis (hPCA) [48] of the combined genomic sparse vectors of multiple species, allowing for distance calculations in n-dimensional space, with the closest shared information content giving the smallest distance using three (or more) components. These first three components can then be used to create a three-dimensional plot of the distances between the sparse vectors for each genome sequence (Fig. 8A). In a similar manner, it is possible to carry out global Boolean analysis on these sparse vectors in a pairwise fashion. By arraying the sparse vectors in an invariant format, with full encoding, it is trivial to carry out pairwise discrete functions for any logic gate functions including [AND, OR, XOR] as seen in Figure 8B. One simple metric is the resultant value of the XOR Boolean operator between two sparse vectors, which we here define as delta tau ($\Delta\tau_k$), or differential occupancy, where k defines the k-mer length. We define the term differential occupancy as the XOR function between two sparse vectors that derives resultant sparse vectors followed by summing the number of non-zero elements and then dividing by total number of possible k-mers, which is 4^k (Chapter II Methods 2.1.3). The more similar the two genomes the smaller the number of resultant non-zero elements after the XOR calculation, resulting in a number that approaches zero for the same species.

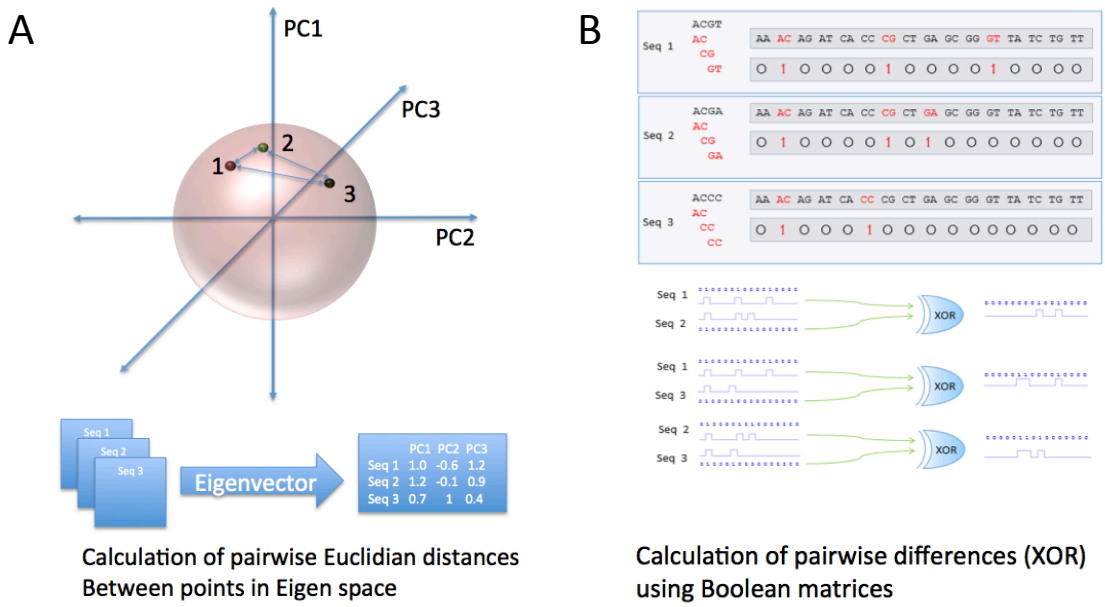


Figure 8 – (A) Linear algebraic forms to analyze a variety of orthogonal transformation / eigenvector calculations. (B) Discrete mathematics enables Boolean analysis using the full complement discrete function XOR.

3.3.4 Clustering of Bacterial sequence

We start by carrying out the hPCA analysis of 578 bacterial genomes in the NCBI bacterial database. We then plot log transformed pairwise distance of these bacterial genomes based on hPCA. By z-transforming the pairwise $-log(\Delta\tau_k)$ data we see a desirable and expected association between species-based similarities between k-mer encoding, and define z>=2 as closely related species. We formalize this association and speculate that this relationship follows a power law association (Fig. 9A), which appears to be inherent in natural genomic topologies of biological systems. We here show punctuated peaks (Fig. 9B) of closely related species within a sea of random or near random associations. Similarly, in calculating genomic topologies of all pairwise $-log(\Delta\tau_k)$ species (Fig. 9C, D) will be defined as those that reside in the k-mer topology

within the exponential function of arrayed pairwise relationships, indicating a "null" and "positive" space. For these closely related species we see r^2 value of 0.87 between the hPCA and Boolean analysis (Fig. 9E), for the species with a pairwise relationship of $z>=2$.

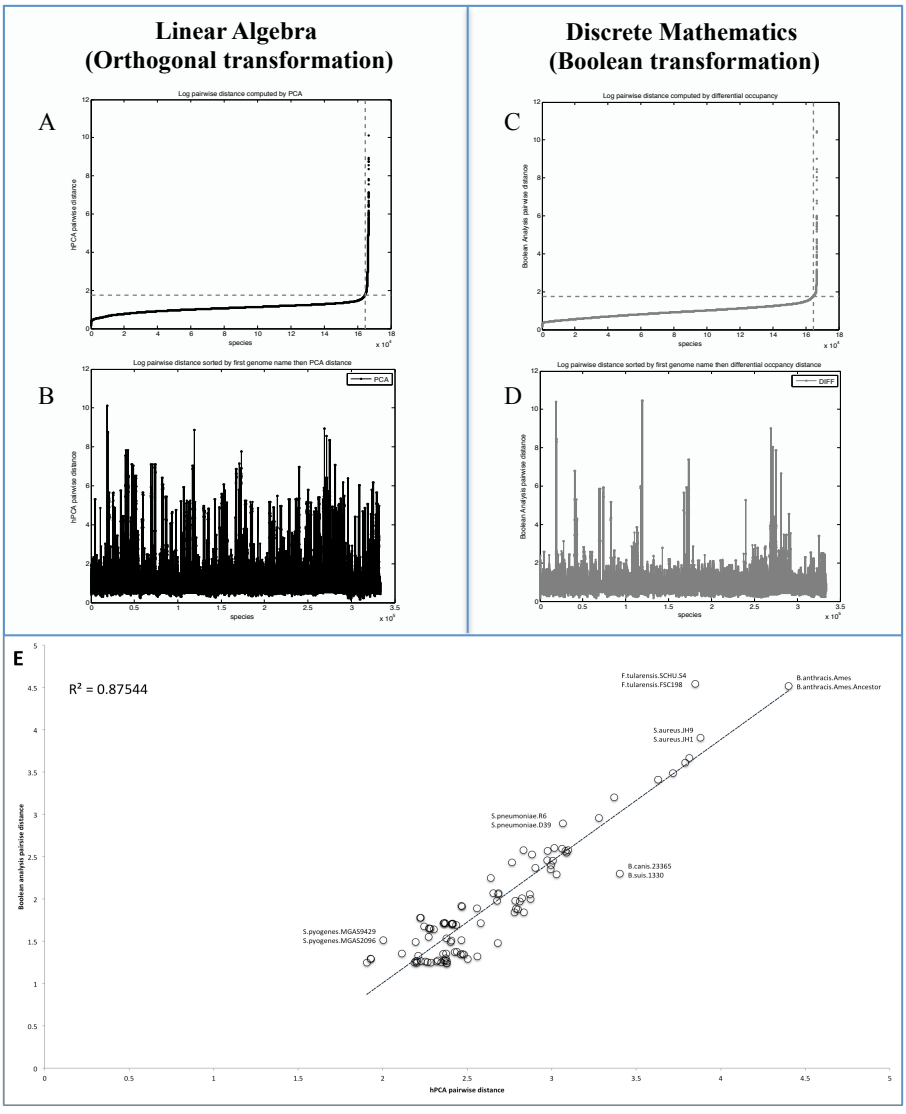


Figure 9 – Pairwise distance. A) Sorted Log pairwise distance computed by hPCA; B) Unsorted Log pairwise distance computed by hPCA; C) Sorted Log pairwise distance computed by Boolean Analysis; D) Unsorted Log pairwise distance

computed by Boolean Analysis; E) Pairwise distance from hPCA and Boolean analysis relationship.

Using this $\Delta\tau_k$ we apply hierarchical clustering and obtain expected clustering of these bacterial genomes (Fig. 10). As previously discussed, we choose *k* equal to 12 and; all 578 bacteria whole genome sequences are k-merized into matrix of integers. Then we build a tree on pairwise distance calculated by Boolean analysis. From Figure 10, species with the same name are closely clustered into the same group. For example, Figure 10 shows a zoomed in portion of the tree where all *Streptococci* are grouped together. Also, *Streptococcus thermophilus and Streptococcus pyogenes* are clustered into two group.

As we present before, K-means, SVM, and hPCA are able to separate bacteria and virus sequences. However, they don't have the ability to separate evolutionarily close species. Also, compared to Boolean analysis, these three algorithms required much more computational resources to finish the analysis of the bacteria and virus data set. The K-means algorithm, in the extreme case, stopped without optimal clustering at maximum iterations (MATLAB 2014b default 100). The overall goal was to create software that could analyze arbitrarily large datasets, and so K-means, SVM and hPCA in practice do not fulfill this goal since they do not scale as well as the Boolean algorithm. The light weight Boolean analysis, as the clustering result shown in Figure 10, enable the possibility for inference of relatedness between viral samples, identification of transmission clusters and sources of infection, which are crucial tasks for viral outbreaks investigations, and finally is able to scale to very large datasets.

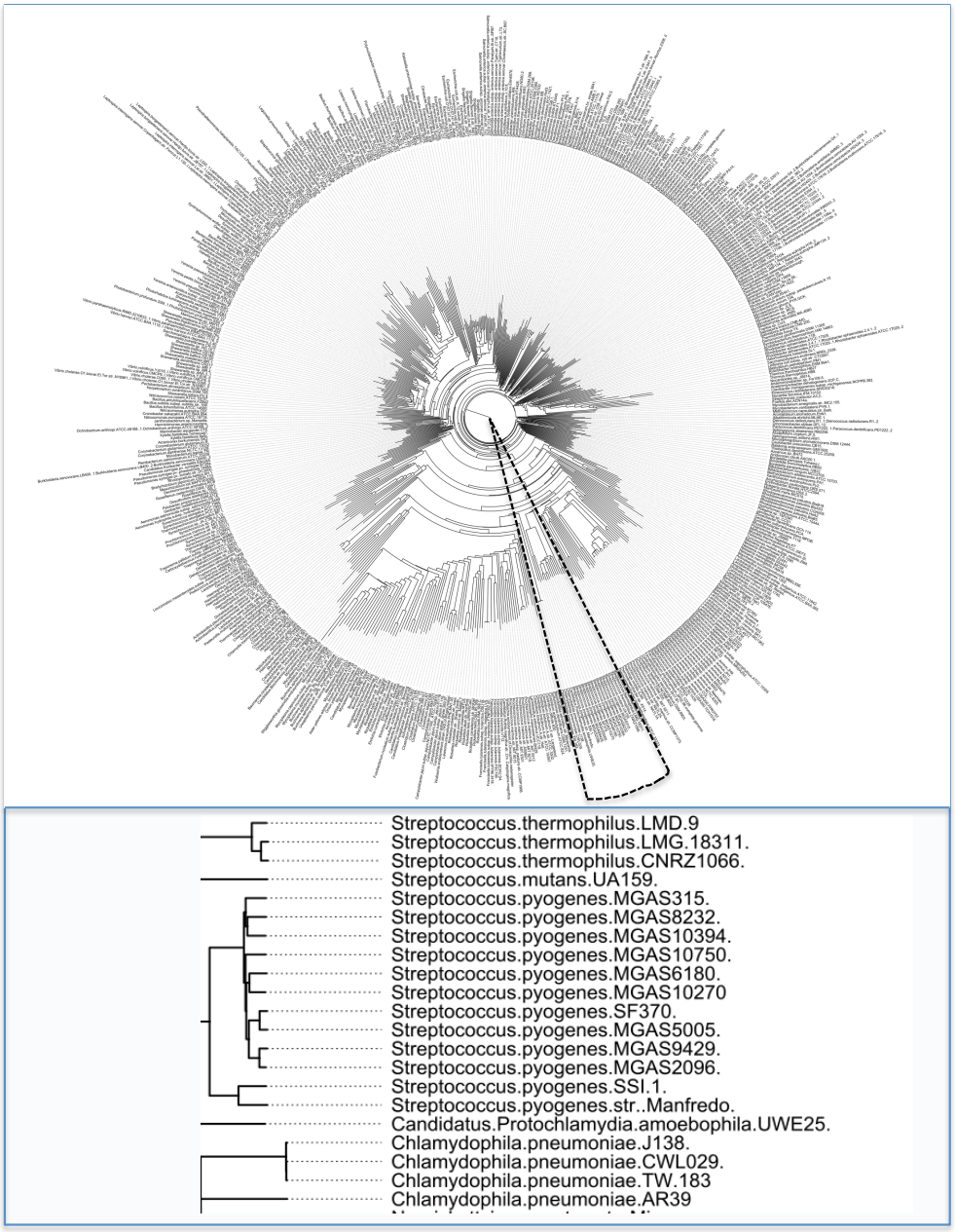


Figure 10 – Hierarchical clustering for bacteria. (A) Complete tree based on pairwise distance, and (B) Zoomed in portion of the hierarchical clustering, generated by differential occupancy.

3.3.5 Clustering of HIV sequences

In another application of Boolean analysis, we performed clustering of HIV sequences. The sequences that we analysed were obtained from the HIV sequence database (https://www.hiv.lanl.gov/content/sequence/HIV/mainpage.html). Based on the origins of the samples, we selected four groups: Asia, Africa, Europe, and America. k-merized sequences, with *k* equal to 12, were used to calculate Boolean analysis pairwise distance. First, we used the data without performing any noise reduction. We then plotted the clustering, using D3 JSON, based on differential occupancy. Figure 11A shows that there is no clear classification among the samples. The reason for this mis-clustering, is that this data was generated from different labs around the world and many of the genomes had variable length regions in the beginning and end. In order to account for this and attempt to remove this issue we applied a noise reduction approach before Boolean analysis. First, we k-merized the sequences and aggregated them into one matrix (termed k-matrix), in which the rows contain k-mers and the columns are samples. Then, we counted the number of times each k-mer was present in the 197 samples. If one of the sample sequences contained the k-mer, we counted it as 1 and the highest count number possible for each row, in this example, was 197. If a k-mer was present in less than 5 samples (i.e. 5/197 or 2.54%), we removed this row/k-mer from the matrix to effectively buffer against noise in this data set. The threshold number 5 is is approximately twice the standard deviation, (i.e. z-score of 2) (Figure 11C). After noise reduction, we applied Boolean analysis, and plotted the clustering based on their pairwise differential

occupancy ($\Delta\tau_k$). Four groups are now separated as shown in Figure 11B, and we can see a much clearer clustering pattern among samples based on their origin.

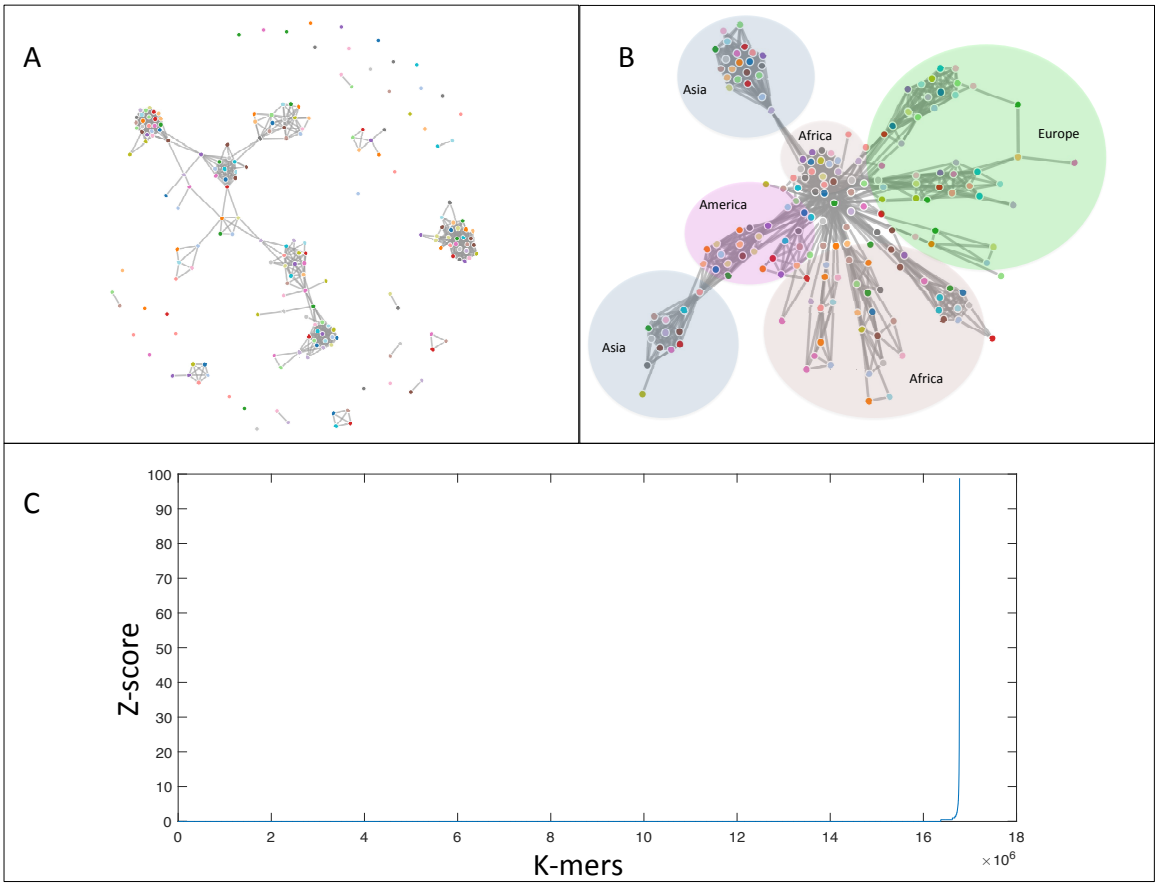


Figure 11 – D3 JSON plot for pairwise distance of HIV genomes generated by calculating differential occupancy. A) plot without noise reduction. B) plot after noise reduction. C) z-score of 12-mers counts among 197 HIV genomes.

3.3.6 Clustering of HCV sequences

There are several studies on HCV outbreaks. One of the most recent successes is Viral Outbreak InferenCE (VOICE) [49-51], which infers genetic relatedness and transmission clusters. We applied our Boolean analysis algorithm on a benchmark HCV data set they provided, which is a collection of HCV intra-host populations sampled from

335 infected individuals. The data consist of 335 intra-host HCV populations, including 142 populations from 33 outbreaks reported to CDC in 2008–2013 and 193 populations from infected individuals without any known epidemiological relationship, all obtained from national collections and other research projects [49]. Each outbreak collections contain from 2 to 19 samples. For all samples, HCV hypervariable region 1 (HVR1) was sequenced. All viral sequences represent a fragment of the E1/E2 genomic region of length 264 bp. Viral populations from two samples are genetically related if they belong to the same outbreak and unrelated, otherwise. The genetic relatedness is validated on the union of collections containing all outbreaks and unrelated samples. There are 24,833,479 pairs of samples, and 99,029 of them are related. We utilized our algorithm to distinguish related versus unrelated HCV sequences and compared our results to the VOICE algorithm. We obtained 95.38% accuracy, 91.70% sensitivity, and 95.39% specificity for the detection of related versus unrelated sequences in a pairwise comparison (Fig. 12). This accuracy is substantially improved over the previously reported <93% accuracy of the VOICE algorithm [3].

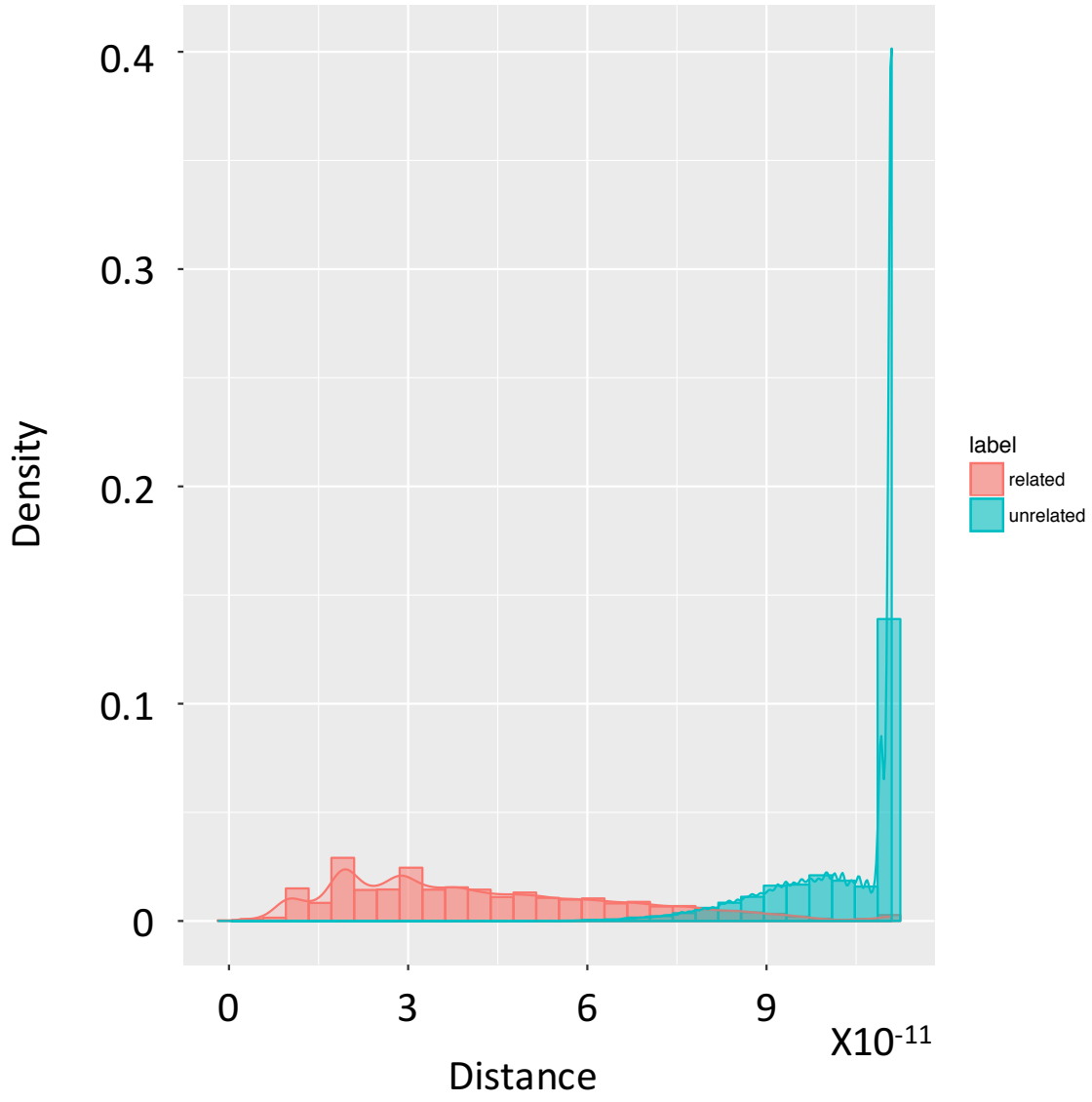


Figure 12 – Histogram of the Finch HCV analysis.

3.3.7 Clustering of Plasmodium malariae and P. ovale genome sequences

We applied our Boolean analysis on a classification of *Plasmodium malariae* and *P. ovale* genomes. A reference genome of *P. malariae* was produced from clinically isolated parasites and sequenced using long-read sequencing technology [52]. The

assembly surpasses available draft genome data for *P. malariae* [53], especially in terms of contiguity allowing large-scale structural changes to be accurately determined. Additional draft genomes for both species of *P. ovale* [54] were assembled from *P. falciparum* co-infections and the genome of *P. reichenowi*, which is a chimpanzee-infective species.

Boolean analysis pairwise distance is calculated on k-merized sequences, with k = 47. As shown in Figure 13, *P. malariae* is separated from *P. ovale*. Also, two mouse-infective species, *P. berghei* and *P. chabaudi* are grouped into one branch away from other species. The most interesting finding is human-infective specie *P. falciparum* and chimpanzee-infective specie *P. reichenowi* are classified together. To investigate host-specific adaptation of parasites to human and chimpanzee hosts, we compared *P. malariae* to *P. malariae*-like and we found lower levels of nucleotide diversity in the human-infective species than in the chimpanzee-infective species. This mirrors the lower levels of nucleotide diversity in the human parasite *P. falciparum* than in its chimpanzee-infective relative *P. reichenowi*. In both cases, the lack of diversity in human-infective species suggests recent population expansions. There is a study using additional samples to calculate standard measures of molecular evolution, they are able to identify a subset of genes under selection in both *P. malariae* and *P. malariae*-like and in an earlier study of *P. falciparum* and *P. reichenowi*, showing some conservation of selection pressures in Plasmodium lineages and suggesting host- specific adaptation of parasites to human and chimpanzee hosts.

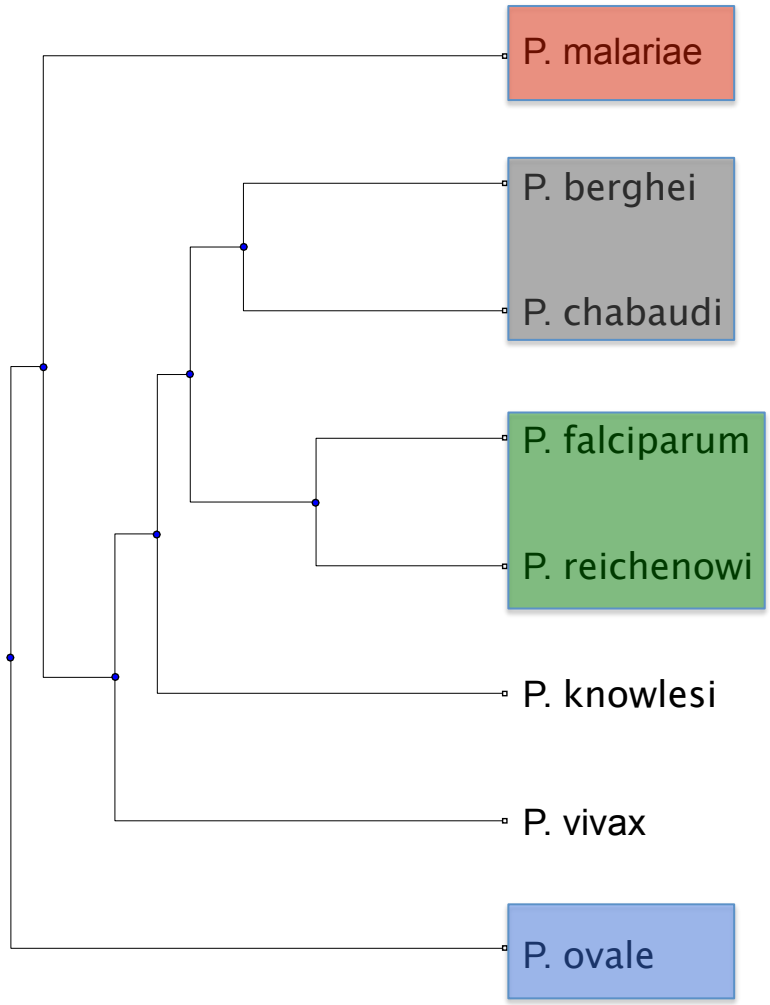


Figure 13 – Hierarchical clustering for Plasmodium species. Hierarchical clustering was generated by differential occupancy for Plasmodium whole genome sequence.

3.3.8 Clustering of Non-O157 STEC whole genome sequences

We applied Boolean analysis algorithm on whole genome sequencing of non-O157 Shiga toxin-producing *Escherichia coli* (STEC) strains [55, 56]. In the United States, according to CDC (https://www.cdc.gov/ncezid/dfwed/pdfs/national-stec-surveillance-overiew-508c.pdf), an estimated 168,698 non-O157 infections occur each year, which is more than the number of cases (estimated at 96,534) caused by STEC

O157. Among the non-O157 STEC strains, serogroups O26, O111, and O103 are considered the most clinically important and frequently identified non-O157 STEC strains in severe diseases and food-borne outbreaks. In Meng's study [57], seven housekeeping genes (*aspC, clpX, fadD, icdA, lysP, mdh, and uidA*) extracted from genomes were selected for multi-locus sequence typing (MLST) analysis. They exclude strains 4865/96 (O145:H28), 101-1 (O?:H10), 5.0959 (O121:H19), DEC12B (O111:H2), and E110019 (O111:H9) because at least one of the selected gene alleles was either absent or only partially present. Here, instead of extracting genes, our program uses whole-genome sequence. We analyzed all strains available, and generated a similar tree by clustering on our Boolean analysis pairwise distance. k-mer files were generated by Kanalyze[19], and utilized the parameter $k = 47$ for this analysis. As Shown in Figure 14, O111:H11, O111:H8, O26:H11 and O103:H2 are separated into different branchs in our clustering dendrogram.

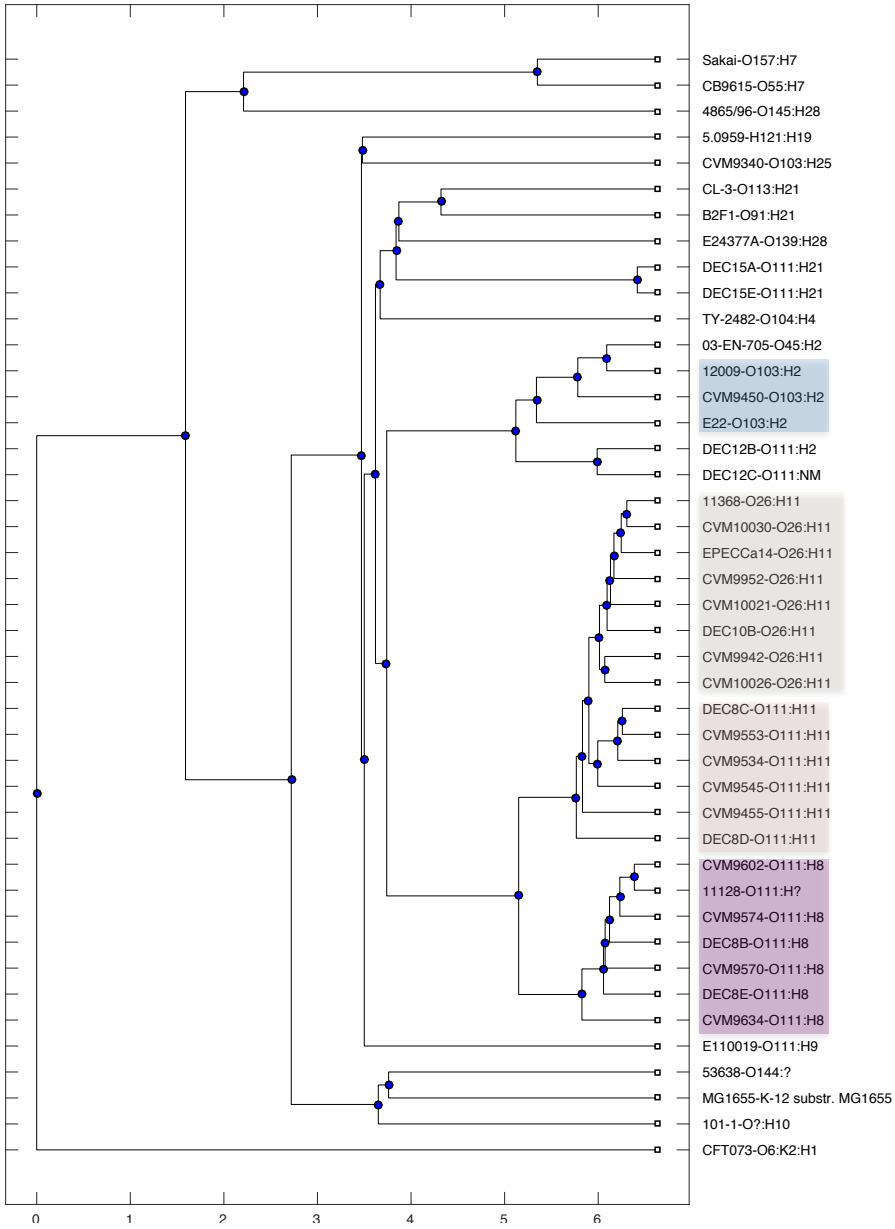


Figure 14 – Hierarchical clustering generated by differential occupancy for non-O157 E. coli whole genome sequence.

3.4 Discussion

Historically it was not always possible to adequately compare complete genomic content between two species dues to the inability to align dissimilar sequences. We

designed a universal system, termed genomic topology in this dissertation, that enables any two sequences to be compared in a fair and simple way, and allows for very different sequences to obtain a definite pairwise score. Using our analysis it does indeed appear that there is a specific power law effect that describes the overall genomic topology between species, with a sharp exponential decay in genomic topology between two different species. This, again, has been suggested but we feel that our result is one of the clearest attempts to show such a genomic topology.

We demonstrate the ability to cluster similar species from first principles using genomic topology calculations without additional levels of abstraction. We show that optimized discrete calculations on global k-mer space qualitatively work as well as traditional linear algebra-based orthogonal transformations eigen value-based principle component analysis, and in certain use cases better than these other algorithms. Although laboratories can and will use analysis of higher levels of abstraction such as the case of PCA and multiple dimensional scaling, it is our opinion that it is easier to intuit intellectually discrete calculations of these higher order matrices verses considering the eigen vector space. An important distinction, which is always troubling with PCA and MDS types of analyses is that the origin of the variation is lost in the calculation, with the inability to determine the origin of the variation. Discrete systems do not face such restrictions, and our system creates the ability to real time cluster and reveals the origin of the variation at the same time, creating a use case that is relevant to pharmacological development. It is not lost on us that a similar encoding system using k-mer/n-tuples can also be carried out for amino acid sequences encoded within organisms, and the parity

between such genomic and proteomic data will also be useful in comparative analysis of species. Indeed, philosophically, this brings to the fore arguments for such numerical taxonomy approaches whereby additional data streams can be added into such analysis, allowing for quantitative assessments of species, including morphological, developmental and other encoding. It should be noted that this is not closed to inter-species analysis, indeed use cases abound that make intra-species analysis informative. Examples include analysis of virulent *E. coli* strains, or viral outbreaks scenarios. Our laboratory is working closely with multiple Centers for Disease Control branches to implement such analysis to allow for near complete automation of such pairwise ($\Delta\tau_k$) analytics during outbreak scenarios.

CHAPTER 4. OPEN SOURCE MACHINE LEARNING ALGORITHMS FOR THE PREDICTION OF OPTIMAL CANCER DRUG THERAPIES

4.1 Abstract

Precision medicine is a rapidly growing area of modern medical science and open source machine-learning codes promise to be a critical component for the successful development of standardized and automated analysis of patient data. One important goal of precision cancer medicine is the accurate prediction of optimal drug therapies from the genomic profiles of individual patient tumors. We introduce here an open source software platform that employs a highly versatile support vector machine (SVM) algorithm combined with a standard recursive feature elimination (RFE) approach to predict personalized drug responses from gene expression profiles. Drug specific models were built using gene expression and drug response data from the National Cancer Institute panel of 60 human cancer cell lines (NCI-60). The models are highly accurate in predicting the drug responsiveness of a variety of cancer cell lines including those comprising the recent NCI-DREAM Challenge. We demonstrate that predictive accuracy is optimized when the learning dataset utilizes all probe-set expression values from a diversity of cancer cell types without pre-filtering for genes generally considered to be "drivers" of cancer onset/progression. Application of our models to publically available ovarian cancer (OC) patient gene expression datasets generated predictions consistent with observed responses previously reported in the literature. By making our algorithm

"open source", we hope to facilitate its testing in a variety of cancer types and contexts leading to community-driven improvements and refinements in subsequent applications.

4.2 Introduction

The sequencing of the human genome, genome-wide association studies (GWAS), quantitative trait loci (QTL) mapping, and similar research initiatives over the past few decades have greatly increased our understanding of the molecular pathways associated with human diseases. These efforts have significantly benefited from the liberal sharing of data and open-source scripts utilized for these efforts. Over the last few years, there has been a number of alternative machine-learning (ML) approaches employed in personalized cancer drug prediction, each associated with variable degrees of success [10, 58, 59]. For example, pRRocphetic [11] is a recently designed R package designed to run the entire learning and subsequent calling of patient data. Other recent contributions include the Bioconductor [60] package SCAN that allows for single-sample array normalization for precision medicine workflows. While a number of ML applications for precision medicine have benefited from community assessments of predicted drug response [e.g., [10, 58]), such efforts have not always shared code, and for the majority of efforts only the organizers of the community assessment exercise were able to see the source code to evaluate each independent solution. This is unfortunate because the open sharing of code has been demonstrated to be a significant catalyst in the optimization of ML applications as in the Large Scale Visual Recognition Challenge (LSVRC) where computational solutions are openly available [61, 62].

We present here an open source software platform using a highly versatile support vector machine (SVM) algorithm that utilizes standard recursive feature elimination (RFE) methods to predict cancer drug response. In pilot applications, we utilized publicly available datasets from NCBI Gene Expression Omnibus (GEO) [63] that we formatted and the array files were partitioned into learning sets and experimental sets. Each individual array is accessible as a .CEL file with individual identifiers at a publically accessible GitHub site that outlines the learning, validation and test sets employed in our initial studies (https://github.com/chuang95/KEA_DrugResponse). Also available at this GitHub site are general procedures for open application of our software to additional datasets (https://github.com/chuang95/KEA_DrugResponseLearning). We have employed the algorithms to explore the effect of a variety of alternative learning datasets on predictive accuracy leading to several unanticipated findings. First, predictive accuracy was significantly improved when microarray probe level expression data rather than average gene expression values were employed in the model building process. Second, predictive accuracy was improved when models were built upon a diversity of cancer types. Third, the pre-filtering of learning datasets based upon preconceived biological models significantly reduces predictive accuracy. Application of our optimized models to publically available ovarian cancer (OC) patient gene expression datasets generated predictions highly consistent with observed responses to a variety of drugs. By providing true open access to our software, we seek to encourage additional improvements in current methods, as well as, constructive comparisons with alternative

approaches leading to the development of optimal ML-based strategies for personalized cancer medicine.

4.3 Result

4.3.1 Support Vector Machine (SVM) model building and recursive feature selection algorithm

A variety of ML techniques and strategies have been employed in the quest for optimal accuracy, sensitivity and specificity in drug response predictions. In this work, we utilize an SVM approach paired with recursive feature elimination (RFE). SVM has been successfully applied in a variety of biological applications in recent years (e.g., [64]). Our SVM models were built using gene expression (https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE32474) (Supplementary table 1) and drug sensitivity profiles (https://wiki.nci.nih.gov/display/NCIDTPdata/NCI-60+Growth+Inhibition+Data) (see also Supplementary table 2) of the NCI-60 panel of human cancer cell lines. Predictive models were built for seven drugs often employed in the treatment of ovarian cancer (carboplatin, cisplatin, paclitaxel, docetaxel, gemcitabine, doxorubicin, gefitinib). The drug sensitivities (GI-50) of the NCI-60 cell lines approximate a normal distribution (Fig 15A; Fig 30). For our learning dataset, we conservatively excluded cell lines displaying GI-50 values within ±0.50 SD of the mean. The test dataset, however, was selected from all cell lines. In all cases, cell lines used to build the models were distinct from those used in testing the models.

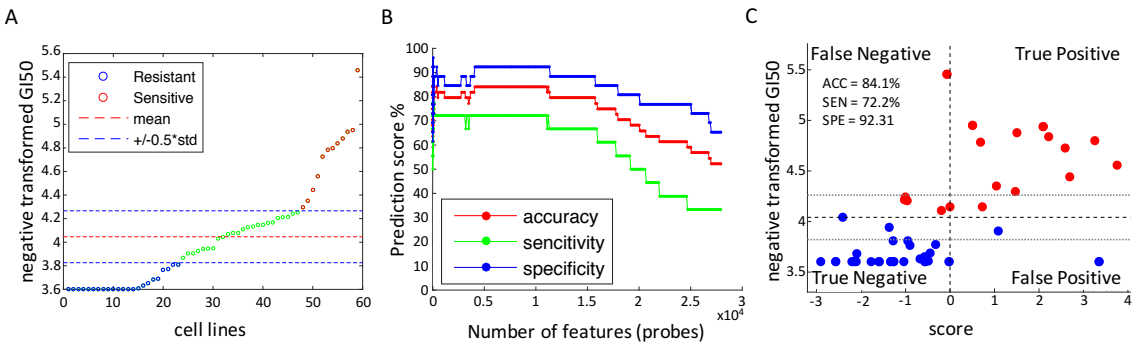


Figure 15 – An SVM-RFE predictive model of carboplatin sensitivity for NCI-60 cell lines. (A) Ranked display of -log transformed GI-50 values for carboplatin for each of the NCI-60 cell lines. Blue circles = carboplatin resistant cells; red circles = carboplatin sensitive cell lines. Cell lines with GI-50 values within ±0.5 SD of the mean (green circles) are less reliably classified as resistant or sensitive and were, thus, not employed in learning datasets. Test sets were selected from cell lines across the entire distribution; (B) Evolution of accuracy of predicted response to carboplatin using SVM-RFE selection for gene probe classifiers; (C) Visualization of the optimal separation between carboplatin sensitive and resistant NCI-60 cell lines. The X-axis is the optimal weight vector (prediction score) of the SVM model for carboplatin; the Y-axis is the -log transformed GI-50 values for carboplatin.

SVM models built upon large datasets typically contain uninformative features, and a number of feature selection methods have been developed to identify subsets of features with optimal predictive accuracy [65-67]. We employed a previously described RFE [68, 69] method to select for features (gene probe sets) that optimally distinguish cells predicted to be sensitive to a drug from those that are not. The RFE method starts by discarding the least relevant features of the model from the bottom of the sorted feature list (Supplementary table 3). The subsequent SVM model is built on the remaining features and again, features with the lowest weights are removed. This process proceeds in a recursive manner until a minimal subset of features is identified that is essential to maintain optimal predictive accuracy. For example, Figure 15B depicts the evolution of predictive accuracy using SVM-RFE feature selection for increased sensitivity to

carboplatin (see Fig 31 for feature selection of the other drugs). In this case, initial removal of uninformative features increased accuracy due to the elimination of features that negatively interact with predictive accuracy. Our SVM-RFE approach compares favorably with other commonly employed methods of feature selection (see Fig 31).

The minimal number of informative features associated with optimally predicted responsiveness to the seven drugs modeled in this study ranged from 10 to 32 (Supplementary table 4). While the biological contribution of the majority of these genes to drug responsiveness is currently unknown, potentially informative trends are often apparent. For example, several of the most informative genes predictive of carboplatin sensitivity have been directly or indirectly implicated with apoptosis (Supplementary table 5), a cellular function known to be induced in response to carboplatin treatment [70].

The SVM models generate drug prediction scores for each cell line. Scores higher than "0" indicate a predicted sensitive response, less than "0" a predicted resistant response (e.g., see Fig 15C, X-axis). The overall accuracy, specificity and sensitivity are evaluated by leave-one-out cross-validation (LOOCV). The SVM computed predictive scores are plotted against observed GI-50 values to graphically display the accuracy of each model. For example, the quadrant plot for carboplatin (Fig 15C) shows that the SVM model is 84% accurate across the NCI-60 test dataset. The predictive accuracies of each of the seven models ranged from 75% to 85% (see Fig 32 for predictive accuracies of the other 6 chemotherapeutic drugs).

4.3.2 Building SVM-based models across a variety of cancer types improves predictive accuracy

While feature selection methods are designed to identify the most informative features by systematically eliminating less informative ones, the predictive accuracy of ML models is heavily dependent on the presumption that the original learning dataset encompasses the full spectrum of features potentially relevant to the predicted variable [10]. The selection of appropriate learning datasets for building predictive models of cancer drug response is especially challenging because a full understanding of the molecular processes underlying cancer onset/progression has yet to be attained [71]. For this reason, subjective limitations in the scope of data employed in learning datasets may negatively affect predictive accuracy of derived models if informative features are inadvertently excluded. For example, it is frequently assumed that models designed to predict optimal therapies for a particular cancer type should appropriately be built using learning datasets derived exclusively from that same type of cancer. However, a growing body of evidence indicates that the molecular pathways underlying cancer onset/progression are not necessarily defined by a tumor's tissue of origin [72]. Thus, a gene expression pattern associated with a particular cancer type may underlie cancer development in other cancer types as well.

To explore this issue, we compared the relative accuracies of two SVM-derived models designed to predict response to the commonly prescribed cancer drug carboplatin. The respective models were built using gene expression profiles and drug response profiles (i.e., learning datasets) derived from 18 of the NCI-60 cell lines. In one case, the

18 cell lines were representative of only two types of cancers (lung and melanoma) while in the other case, the 18 cell lines were randomly selected to be representative of all 9 types of cancer comprising the NCI-60 dataset (lung, colon, breast, ovarian, leukemia, renal, melanoma, prostate and CNS). As shown in Figure 16A, the model built using data from the 9 cancer types was more accurate in predicting carboplatin sensitivity (87.5%) than the model built upon only two cancer types (75.0%) (Fig 33). This finding is consistent with growing evidence that the molecular basis of individual cancers may not necessarily be defined by tissue of origin [17]. In addition, the fact that variation in gene expression levels is typically greater among multiple cancer types (Fig 34) may be an additional relevant factor since the predictive accuracy of ML models is well known to improve with increasing diversity of the learning set data [10].

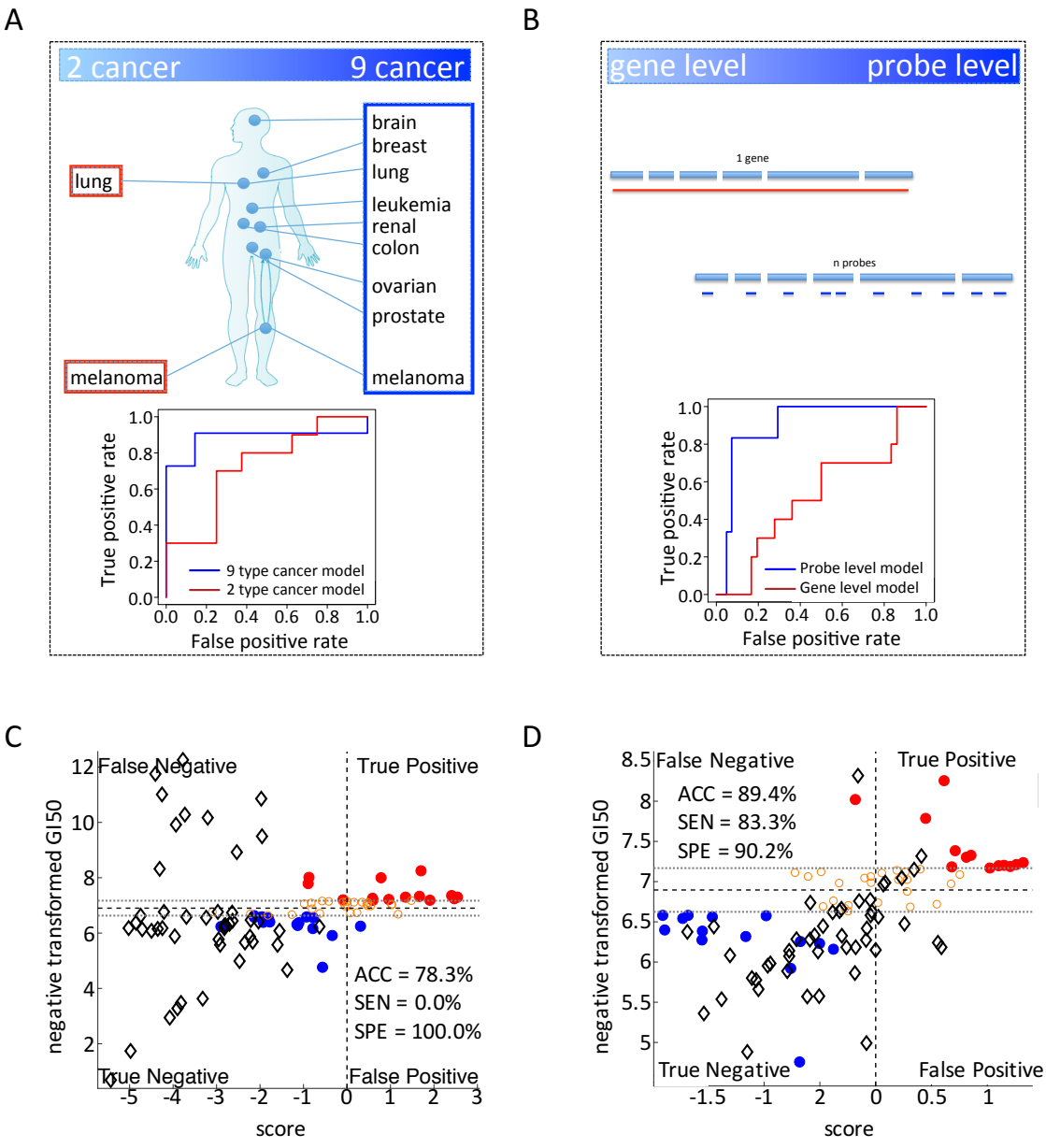


Figure 16 – The influence of learning datasets on the predictive accuracy of SVM-RFE models. (A) Comparison of predictive accuracy (ROC curves) for two SVM models of response to carboplatin using a learning dataset derived from 2 cancer types (lung, melanoma) vs. 9 cancer types (brain, breast, lung, leukemia, renal, colon, ovarian, prostate and melanoma). In each case, the data were derived from a total of 18 cell lines. The results indicate that the model built using learning set data from 9 cancer types generates a more accurate prediction (see also Fig 33); (B, C, D) Prediction of the sensitivity of breast cancer cell lines to doxorubicin. In one case, the model was built using a learning dataset comprised of average gene expression values. In the other case, the model was built using a learning dataset comprised of

the expression values of all gene probes. The results demonstrate that the model built using probe set data is more accurate than the model built using average gene expression data; (C) prediction score accuracy using average gene expression values; (D) prediction score accuracy using expression values of all gene probes (Red circles = drug sensitive training set; Blue circles = drug resistant training set; Black diamonds = breast cancer cells test set).

4.3.3 The averaging of microarray probe set expression values reduces predictive

Another way in which the information content of learning datasets may be compromised is by the employment of average rather than raw experimental values. For example, Affymetrix and other microarray gene expression systems typically incorporate multiple probe sets per gene, thereby providing the possibility of monitoring differences in levels of alternative splicing and other post-transcriptional expression variants (e.g., Fig 34). While the use of average gene expression values may be appropriate for many applications, the loss of information associated with the use of such average values in learning datasets could negatively affect the accuracy of drug prediction algorithms if, for example, rare splice variants turn out to be particularly informative features.

To test this possibility, we compared the relative predictive accuracies of two SVM-based algorithms developed to predict the sensitivity of the set of breast cancer cell lines recently employed in the NCI-DREAM Challenge to the drug doxorubicin [1]. In one case, we employed the average Affymetrix gene expression dataset that was provided to the Challenge participants (https://www.synapse.org/#!Synapse:syn2785783). In the other case, we downloaded and employed the original probe data as our learning set (ArrayExpress E-MTAB-181, http://www.ebi.ac.uk/arrayexpress/experiments/E-MTAB-181/). The results presented in Figure 16B, C and D demonstrate that the model built

using the probe set data is substantially more accurate (89%) in predicting the sensitivity of the breast cancer cells lines to doxorubicin than the model (78%) built using the averaged gene expression values.

4.3.4 Pre-filtering of learning datasets can reduce predictive accuracy

Some methods to assess risk of cancer progression and/or severity focus almost exclusively on genes previously identified as drivers of cancer onset/progression [73]. The advantage of such data pre-filtering is a reduction in the complexity of downstream analyses but, as discussed above, it may also negatively impact the accuracy of derived predictive models if the truncated datasets do not encompass all genes associated with drug sensitivity.

To explore this question, we compared the predictive accuracy of two SVM-based models using the above breast cancer cell line data. In one model, the learning dataset consisted of expression patterns of 297 genes previously implicated in cancer onset/progression [74] (http://foundationone.com/docs/FoundationOne_tech-info-and-overview.pdf). In the second model, the learning dataset included probes of all significantly expressed genes (Supplementary table 1). The models built using the pre-filtered data from the 297 genes were substantially less accurate (59.6%) in predicting responses to doxorubicin than the model built upon unfiltered data (89.4%) (Fig 17).

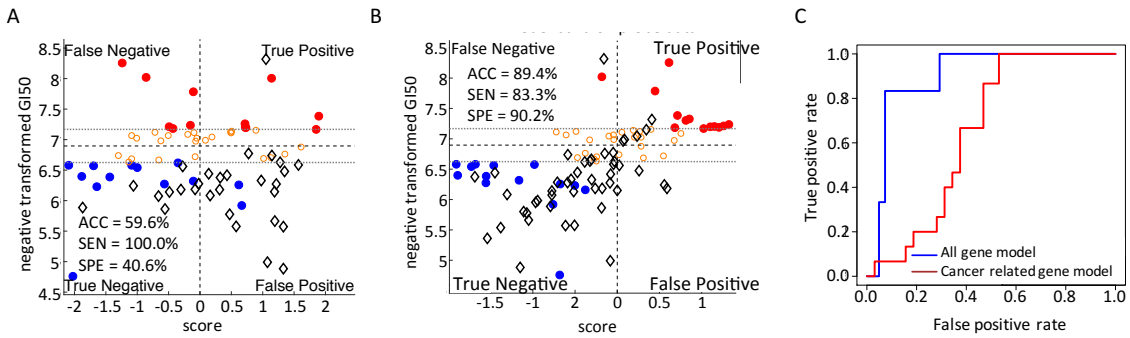


Figure 17 – Pre-filtering of learning datasets can reduce the accuracy of predictive models. Shown is the predicted sensitivity of breast cancer cell lines to doxorubicin by two SVM models built using different learning datasets. In one case, the model was built using a learning dataset limited to the expression of 297 genes previously associated with cancer onset/progression [19]. In the other case, the model was built using a learning dataset drawn from all significantly expressed genes (Supplementary table 1). The results indicate that pre-filtering of the learning dataset to only include gene expression values of previously identified cancer related genes reduces predictive accuracy. (A) Quadrant plot of SVM predicted sensitivity to doxorubicin vs. observed sensitivity to doxorubicin of model built using a learning dataset pre-filtered for genes previously associated with cancer onset/progression; (B) Quadrant plot of SVM predicted sensitivity to doxorubicin vs. observed sensitivity to doxorubicin of model built using all gene expression data (Supplementary table 1); (C) ROC curves of the two models showing reduced predictive accuracy associated with the pre-filtered learning dataset (Red circles = drug sensitive training set; Blue circles = drug resistant training set; Black diamonds = breast cancer cells test set).

4.3.5 Model applications to human cancer datasets

While our predictive models were established using gene expression and drug sensitivity data from human cell lines, we were interested in conducting preliminary evaluations of the models' ability to predict the response of human cancer patients to chemotherapeutic treatments. Toward this end, we downloaded three independently derived (Affymetrix) gene expression datasets of 273 ovarian cancer patient tumors from the Gene Expression Omnibus (GEO) repository (GSE30161, GSE18521, GSE20565;

http://www.ncbi.nlm.nih.gov/gds). The expression values for each individual array were normalized back to the NCI-60 gene expression data matrix.

Using these data, we employed our models to predict the response of the 273 cancer patients to cisplatin, doxorubicin, paclitaxel, carboplatin, docetaxel, gemcitabine and gefitinib. For example, Figure 18A and B display the predicted response of two randomly selected patients from the GEO data set. One of the patients (Fig 18A) is predicted to respond favorably to the standard first-line therapy (carboplatin/paclitaxel) while the second patient (Fig 18B) is not. Interestingly, the patient predicted not to respond to first line therapy, is predicted to respond favorably to gemcitabine. Unfortunately, the observed response of these individual patients to therapy is not available. However, the collective response of ovarian cancer patients to the seven drugs analyzed in these studies has been previously reported (Supplementary table 6). To compare the collective predictive accuracy of our models to the collective observed response rates, we combined the predictive sensitivities of the 273 patients comprising the 3 GEO datasets and displayed the results as a distribution of the combined SVM predicted scores (Fig 18C). The results indicate that while at least some patients are predicted to respond to each of the seven drugs, the vast majority (75%) of patients are predicted to respond favorably to carboplatin (Fig 18C), followed closely by gemcitabine, cisplatin (58%) and paclitaxel (56%). Of interest is the fact that carboplatin, given concurrently with paclitaxel, is the current first-line chemotherapy for ovarian cancer patients, with approximately 75-80% of patients being responsive to this combination [75]. Our predictions suggest that the drug primarily responsible for this favorable

response is carboplatin. Gemcitabine is commonly given as a second-line chemotherapy for OC and has been found to be of moderate clinical effectiveness, in line with our predictive models [76]. Figure 18D displays the linear regression between the predicted response rates to the seven chemotherapeutic drugs by our models and the observed response rate from clinical studies (Supplementary table 6). The overall predictive accuracy of our models in this dataset is > 80% (r^2 = 0.8201).

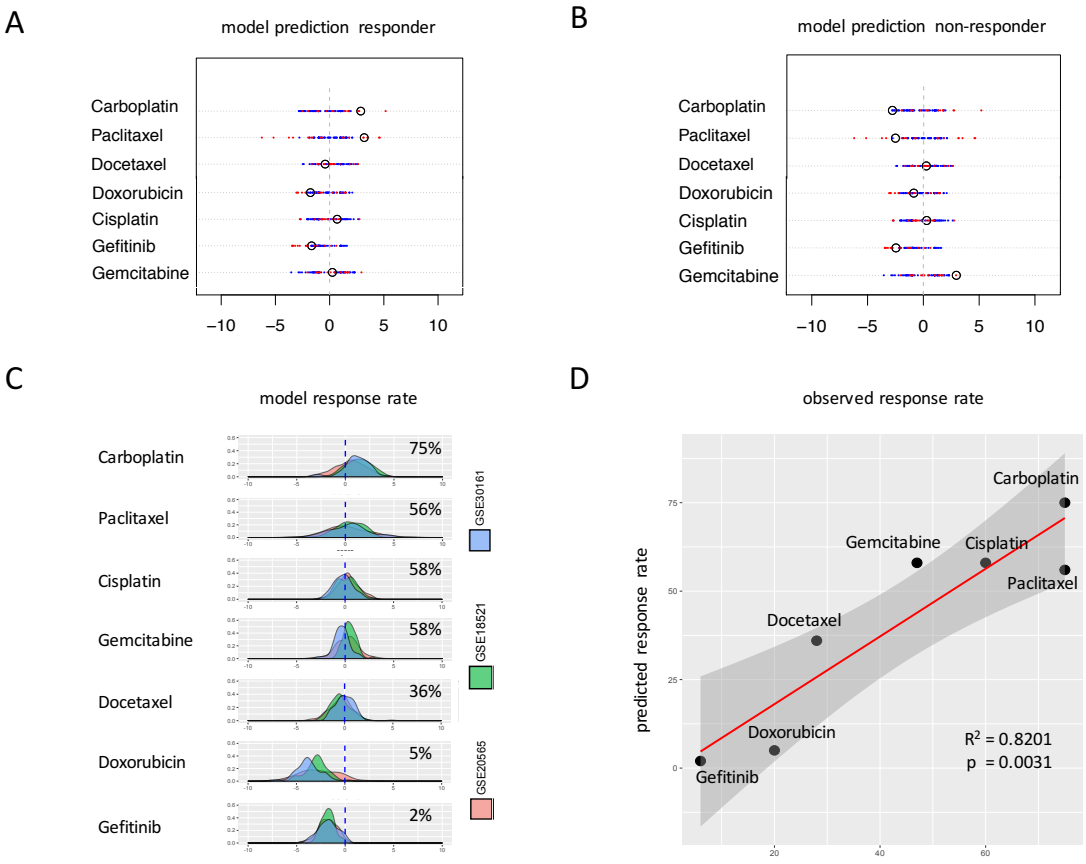


Figure 18 – Individual and aggregate prediction of response to chemotherapeutic drugs. The SVM algorithms output binary classifications for each drug (sensitive/resistant) established through a decision function that numerically separates cancer cells predicted to respond to the drug (positive score) from those predicted to be non-responders (negative score). (A) The predicted response of an individual patient (GSM516724) to seven chemotherapeutic drugs. This patient is predicted to respond favorably to the first line therapies of carboplatin (score 2.88)

and paclitaxel (score 3.20). (B)The predicted response of a second individual OC patient (GSM516801) to seven chemotherapeutic drugs. The patient is predicted NOT to respond favorably to the first line therapies of carboplatin (score -0.28) and paclitaxel (score -2.53). (C) Density plot of aggregate prediction scores for 3 GEO data sets of 273 ovarian cancer patients and the predicted group response rate for each drug. (D) Scatter plot of the predicted group response rates vs. the observed group responses of OC patients to seven chemotherapeutic drugs (Linear regression p value = 0.0031, r^2 = 0.8201) (Supplementary table 6).

4.4 Discussion

A primary goal of personalized cancer medicine is the accurate prediction of optimal drug therapies based upon individualized molecular profiles of patient tumors [77]. In an ideal world, such predictions are based upon firmly established cause and effect relationships between identified molecular aberrations and specific aspects of the onset and progression of the disease. An example is the well-established relationship between constitutively active Bcr-Abl tyrosine kinase (TK) expression and the leukemic phenotype associated with CML (chronic myelogenous leukemia) [78]. Patients identified with this molecular aberration are effectively treated with targeted TK inhibitors that work to reduce the elevated activity and restore regulatory balance to the cell. Regrettably, the underlying molecular causes of most tumors are, as yet, not as well understood as for CML. This has leadled to growing interest in the application of ML approaches to the prediction of optimal drug therapies [73]. ML-based predictive models are not predicated upon knowledge of underlying cause and effect relationships but rather on the identification of significant correlations between specific components of tumor molecular profiles and the favorable response of tumors to specific drugs.

The open source availability of ML prediction algorithms provides the research community with unique opportunities for creative modifications and improvements of existing algorithms not otherwise possible. For example, open sharing of code has been critical to improvements in ML approaches to image recognition [61, 62].

Despite the documented advantages of the open sharing of code, to date, the practice has been extremely limited within the field of cancer drug prediction. For example, there is a notable lack of GitHub, Sourceforge, R Bioconductor and other online repositories of cancer drug prediction applications in contrast to the resources available for other ML applications such as the Large Online Image (LOI) repository competitions where alternative computational solutions are openly deposited [62]. We believe that making cancer drug prediction algorithms open source could result in similar benefits in the field of personalized cancer medicine.

Toward that end, we present here an open access support vector machine (SVM)-based algorithm for the predictive response of cancers to seven widely employed chemotherapeutic drugs. The algorithm combines a standard SVM approach with a "one-by one" data normalization pipeline. We have employed the algorithm to explore the effect of a variety of alternative learning datasets on predictive accuracy leading to several unanticipated findings. For example, although it may seem intuitive that drug predictive models for a specific type of cancer should optimally be built upon data from the same cancer type, our results suggest that this may not always be the case. The predictive accuracy of the drug response of a particular cancer type was significantly increased when the model was built using data from a variety of cancer types. This

finding is consistent with growing evidence that molecular signatures of optimal cancer drug response are not necessarily defined by the cancer's tissue of origin [72].

Microarray platforms typically monitor gene expression levels using multiple probe sets. This allows discrimination between the expression patterns of alternative splice variants and/or other gene transcript isoforms. Most often, the input expression data for the building of ML predictive models utilizes average expression values across all gene probes. We found that higher accuracy is attained when all probes are incorporated in the learning dataset presumably because some isoforms are more informative than others with respect to drug response and this information is lost or diluted when individual probe data are combined in an average value.

Personal cancer drug therapy, as currently envisioned, involves the targeted inhibition of one or more "cancer driver" genes, i.e., genes that have been previously identified as playing key roles in cancer onset and progression. For this reason, the molecular profiles of putative cancer driver genes or other pre-defined subsets of genes are often considered sufficient for the accurate prediction of optimal drug therapies. We found that predictive accuracy can, in fact, be significantly reduced when expression profile datasets are pre-filtered prior to ML-based model building. This result suggests that genes involved in cancer drug response are not necessarily limited to those involved in cancer onset even when the drug in question is designed to target a specific group of driver genes. This also can be referred to over fitting problem. At this point, our model utilized linear kernel support vector machine as cluster and performed feature elimination on our learning set, NCI 60 cell line microarray. The most common way to tune this SVM model is optimizing its boxconstraint parameter *C*. For each step of feature

elimination, we tune the model by try different *C* from range [0.1:0.1:1 2:1:10 20:10:100], and select the best *C* for each step. Figure 19A shows the performance of best model by tuning *C* yields accuracy better than 95%. However, when we test the optimized model on clinical patients (Fig. 19B), all sample are classified into one category, either extreme responder or extreme non-responder. Obviously, the model is over fitted for our learning set, NCI 60 cell lines data set. Over fitting is common problem in machine learning problems, and our learning set is a well-maintained clean tumor cell line data set. To avoid this type of over fitting, we need a larger data set, especially clinical patient microarray gene data.

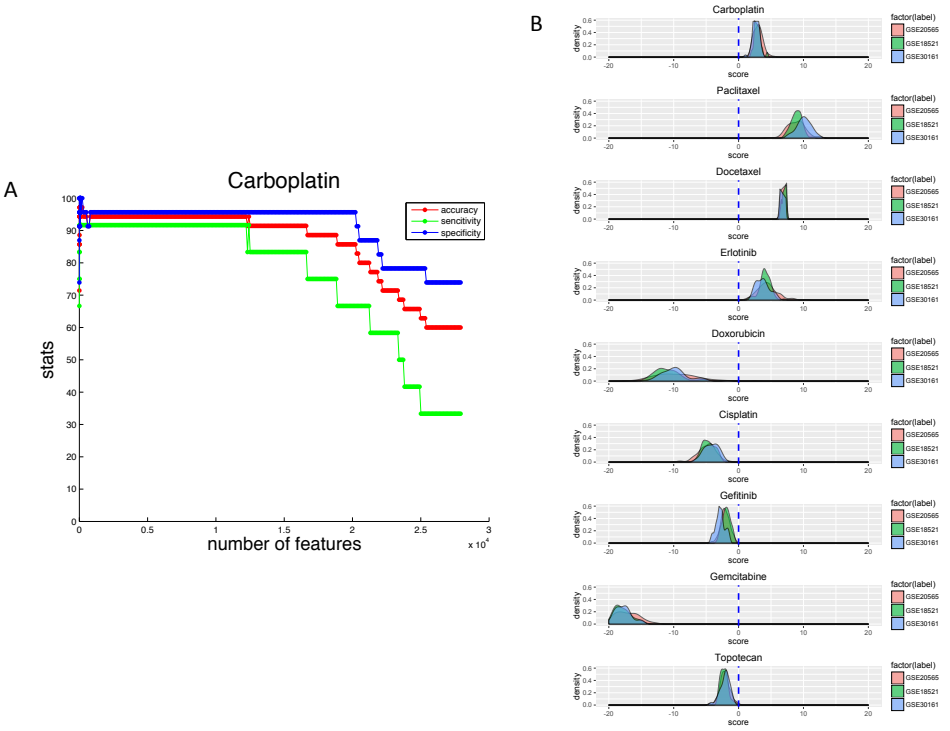


Figure 19 – Model optimization by tuning boxconstraint parameter *C* for SVM. A) Performance of optimizing model on NCI 60 cell lines data set. B) Optimized model test on 3 sets of clinical ovarian cancer patients (GSE20565, GSE18521, GSE30161).

Although our models were built using the publically available NCI-60 cancer cell line datasets, we are encouraged that predictions using publically available human patient datasets are generally consistent with clinical observations. By making our predictive models open source, we hope to encourage the testing of predictions in additional human datasets representative of a diversity of tumor types. we test our model on non-small cell lung cancer (NSCL) patients (GSE19804). Figure 20 shows the density of predicted scores for both NSCL cancer and ovarian cancer patients (GSE30161). By comparing these two together, we can see the response rate of drug Carboplatin for NSCL patient is lower than it for ovarian cancer patients. This indicates that the drug Carboplatin is not effective on threating NSCL cancer patients as it on ovarian cancer patients. According to NCI, this is true. Because Carboplatin is mostly used on ovarian cancer, not NSCL cancer. To move to a new chapter, you must tell Word that you are moving on to a new page. To prove our algorithm can be used for other type of cancer, we are going to build models for more FDA proved drugs that effective on other type of cancer.

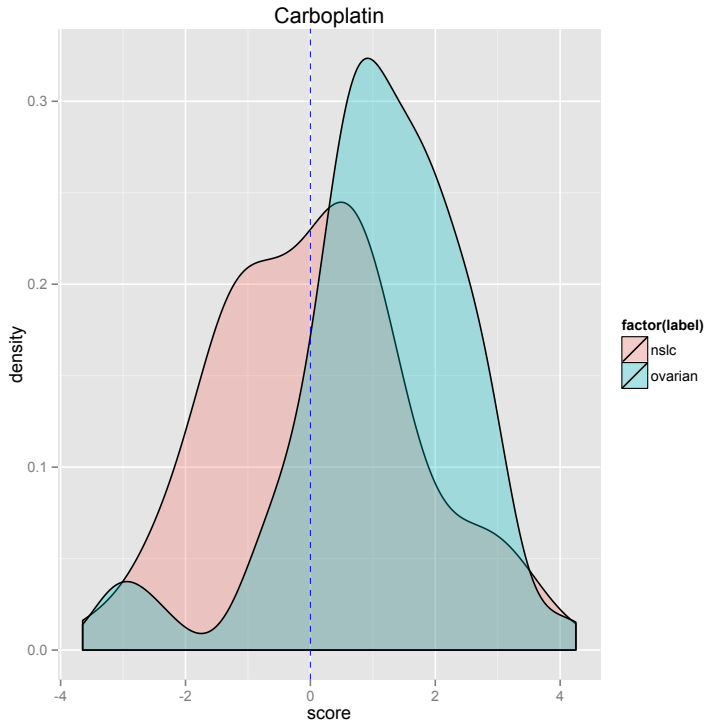


Figure 20 – Model prediction scores on non-small cell lung cancer patients and ovarian cancer patients.

Another approach to optimize our model is building a model for all drugs. We start by test our models for each drug on other drugs in our database. As shown in Figure 21, some models did good job predicting drug response on other drugs. For example, models of Erlotinib and Gefitinib perform well on each other. They both predict over 75% accuracy on each other testing set. That's because they are similar target drugs. However, most models perform poor on other drugs. Different model of drug selected different subset of most informative genes, because drugs target different cancer pathways. One model to rule all drugs seems hard to achieve, but one model with dynamic selection of genes for different combination of therapies maybe the next step of our grogram.

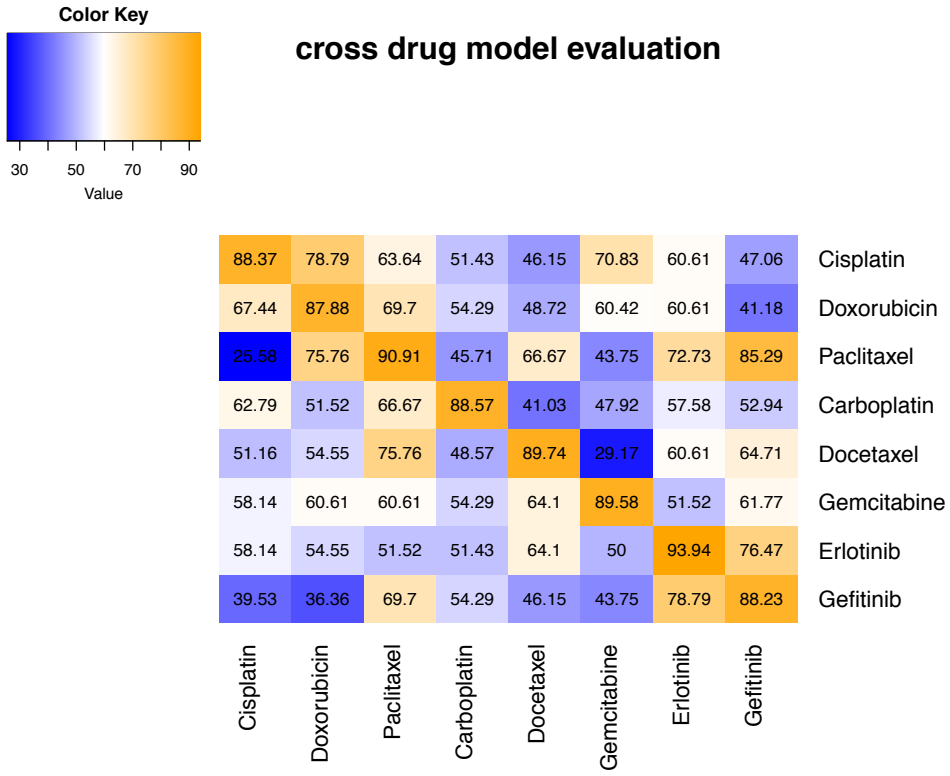


Figure 21 – An SVM-RFE predictive model of carboplatin sensitivity.

In summary, our findings demonstrate that significant improvements can be made in the predictive accuracy of ML-based algorithms by modulating the format and/or type of learning datasets employed in the model building process. This finding is likely to be relevant regardless of the type of ML approach employed. While our results illustrate several paths by which the predictive accuracy of our ML-based cancer drug prediction algorithm was improved, these and additional possibilities need to be tested with larger and more extensive datasets. We believe that such goals are most effectively attained by communal efforts where the research community is provided open access to the underlying code and pipelines employed so that meaningful improvements and

comparisons with alternative methods can be made [79]. Toward this end, we currently provide an open source R package and pipeline for application of our prediction methods (https://github.com/chuang95/KEA_DrugResponse). In addition, a user-friendly web server is currently under construction that will further enhance public access to our methods. It is our hope that through community sharing of this and other open source cancer drug prediction algorithms and associated data formatting/normalization procedures that the attainment of a major goal of personalized cancer medicine will be facilitated.

CHAPTER 5. MACHINE LEARNING PREDICTS INDIVIDUAL CANCER PATIENT RESPONSES TO THERAPEUTIC DRUGS WITH HIGH ACCURACY

5.1 Introduction

A primary goal of precision cancer medicine is the accurate prediction of optimal drug therapies based upon the personalized molecular profiles of patient tumors [80]. Ideally, such predictions are based upon well-established molecular cause-and-effect relationships that are disrupted in cancer cells. A notable example is the targeted inhibition of the Abl tyrosine kinase protein in the treatment of chronic myelogenous leukemia (CML) [81]. Unfortunately, the molecular processes underlying most cancers, and especially solid tumors, are currently not as well understood as for CML [82]. An alternative path to accurate predictions is based simply on observed, highly-significant correlations, even when the underlying causal connections are unknown or incompletely understood.

The foundation of accurate correlative predictions is built upon extensive and reliable bodies of data, and the volume of cancer-relevant data being generated and computationally stored on a daily basis vastly exceeds what could be even imagined only a few decades ago. For example, the volume of cancer-relevant molecular data being generated by genomic studies alone (DNA sequencing, RNA expression, etc.) is currently doubling about every 6-7 months and, within the next decade, is estimated to constitute up to 40 million gigabytes a year [83].

The search for highly significant correlations in cancer-relevant datasets is a task ideally suited to computers and specifically to a branch of artificial intelligence called machine learning (ML). Towards that end, a number of ML-based approaches have been developed in recent years that input the genomic profiles of individual patient tumors and output predictions of optimal drug responses based upon correlations embedded within previously established datasets [84]. We recently introduced open source access to a support vector machine (SVM)-based algorithm that inputs gene expression profiles of cancer cells to predict the response of individual cancers to chemotherapeutic drugs [85]. We previously employed this algorithm to predict the sensitivities of 273 ovarian cancer patients to seven commonly prescribed drugs [85]. These predictions were shown to correlate significantly with previously reported response rates of independent groups of ovarian cancer patients to these drugs (Linear regression p value = 0.0031, $R^2 = 0.8201$). We present here the use of SVM-based algorithms to predict the responses of individual cancer patients to a variety of standard-of-care chemotherapeutic drugs from gene-expression profiles (RNAseq or microarray) of patient tumors (https://github.com/chuang95/KEA_DrugResponseRNA-seq). The accuracies of the models to predict responses to a variety of drugs across 175 patients ranged from 81.5% to 83.6%. The potential clinical utility of our SVM-based approach, particularly with respect to the selection of drugs for patient's resistant to first-line chemotherapies, is discussed.

5.2 Results

5.2.1 The response of individual cancer patients to gemcitabine or 5-fluorouracil therapy is predicted with >80% accuracy

To assess the accuracy of our SVM-based algorithms to predict drug response on an individual patient basis, we first employed matched sets of gene-expression and drug-response profiles from The Cancer Genome Atlas (TCGA) database (TCGA https://cancergenome.nih.gov/). The TCGA database is comprised of 2.5 petabytes of data including the genomic profiles of tumor and matched normal tissues from more than 11,000 patients representing 33 types of human cancers. Despite the impressive size of this dataset, we were limited because we require not only gene-expression profiles of patient tissues but detailed information on each patient's individual response to chemotherapy as well. Since the availability of such correlated sets of data for specific cancer types is currently limited, we combined TCGA data of patients associated with a diversity of cancer types but for which the response profiles to two commonly employed chemotherapeutic agents, gemcitabine (GEM) and 5-fluorouracil (5-FU), have been well documented. In this way, we were able to establish a dataset comprised of expression profiles (RNAseq) and drug response profiles of 152 patients (92 treated with gemcitabine, 60 treated with fluorouracil) (Table 3).

Table 3 – Number and types of cancer patients responding to gemcitabine or 5-fluorouricil chemotheraputic treatments.

Cancer Type	Drug	No. Responsive	No. Not Responsive	Total No. Samples
Bladder	Gemcitabine	4	4	8
breast Invasive	Gemcitabine	1	1	2
cervical	Gemcitabine	1	0	1
cholangiocarci	Gemcitabine	3	1	4
head/neck	Gemcitabine	0	1	1
liver	Gemcitabine	0	2	2
lung	Gemcitabine	1	2	3
lung squamous	Gemcitabine	1	2	3
pancreatic	Gemcitabine	24	33	57
phenochromoc	Gemcitabine	0	1	1
sarcoma	Gemcitabine	1	5	6
skin cutaneous	Gemcitabine	0	2	2
testicular germ	Gemcitabine	1	0	1
uterine corpus	Gemcitabine	0	1	1
Sub-total	Gemcitabine	37	55	92
colon	Florouricine	2	2	4
esophageal	Florouricine	2	1	3
pancreatic	Florouricine	3	6	9
rectum	Florouricine	9	0	9
stomach	Florouricine	19	16	35
Sub-total	Florouricine	35	25	60
		74		

Independent predictive models were built for GEM and for 5-FU utilizing the gene expression and patient outcome data obtained from the TCGA database. Unlike our earlier models that were built using microarray gene-expression data [85], the gene-expression values in the TCGA dataset are recorded as RNAseq profiles. Our model building and testing methods, however, remain essentially as previously described [85].

In the TCGA database, patient responses to drugs are grouped into 4 categories: complete response, partial response, progressive disease and stable disease. Since the current configuration of our algorithms require a binary input with respect to drug response, we classified patients displaying either complete or partial response to the drug treatment as responders (R) and those displaying progressive or stable disease following treatment as non-responders (NR) (Table 3).

The profiles of 75% of the patients (i.e., 69 patients for GEM; 45 patients for 5-FU) were randomly selected to establish the learning datasets for model building and the remaining 25% (i.e., 23 patients for GEM; 15 patients for 5-FU) were employed as the test datasets for initial evaluation of the models.

ML models built from large datasets typically contain uninformative features that can reduce predictive accuracy. For this reason, several feature selection methods have been developed to establish subsets of features with optimal predictive accuracy [67, 86] . In our studies, we employ a recursive feature elimination (RFE) method [85] to select for features (i.e., gene-expression patterns) that can optimally distinguish between predicted responders and non-responders. The RFE method begins by discarding the least relevant

features of the model from the sorted feature list (Supplementary table 7). The subsequent SVM model is built on the remaining features and again, features with the lowest weights are removed. This process proceeds in a recursive manner until a minimal subset of features is identified that is essential to maintain optimal predictive accuracy. Figure 22 depicts the evolution of predictive accuracy using SVM-RFE feature selection for increased sensitivity to GEM and 5-FU. The minimum number of informative features associated with optimally predicted responsiveness to GEM was 81and for 5-FU was 31 (Supplementary table 8).

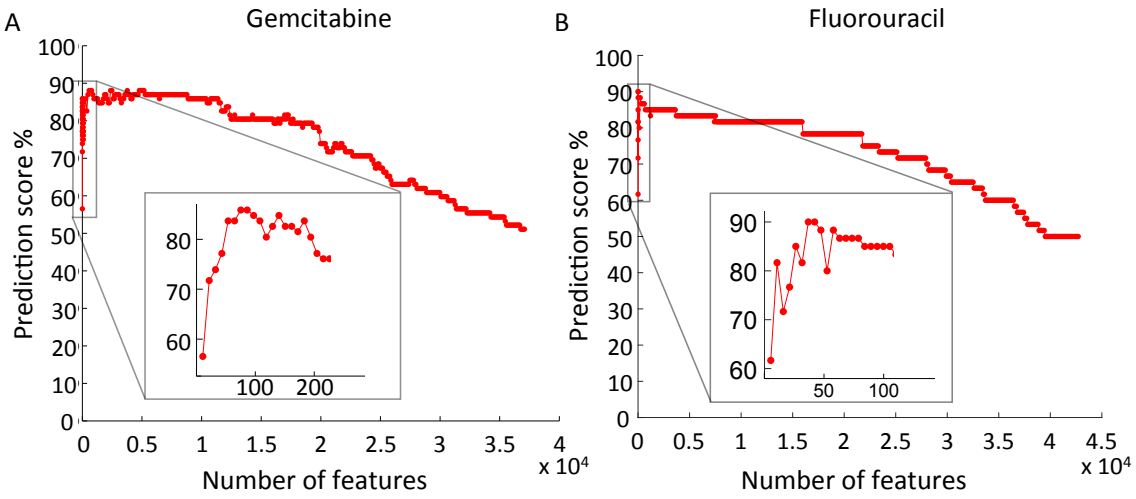


Figure 22 – Evolution of accuracy of predicted response to gemcitibine (A) and 5-Fluorouricil (B) using SVM-RFE selection for gene probe classifiers.

Employing a set of most informative features, the SVM-RFE models generate drug prediction scores for each patient. Scores greater than "0" indicate a predicted positive response to the drug while scores less than "0" are predictive of drug resistance. Figure 23 displays the distribution of prediction scores for the 92 patients treated with gemcitabine and the 60 patients treated with 5-FU (see also Supplementary table 9).

Patients observed to respond positively to the drug therapy are represented in the figures by blue dots and those observed not to respond to the therapy by red dots. The overall accuracies (gemcitabine 81.5%; 5-FU 84.1%), sensitivities (gemcitabine 75.7%; 5-FU 88.6%), and specificities (gemcitabine 85.5; 5-FU 76.0%) of the two models were determined by leave-one-out cross validation (LOOCV) as previously described [85]. The high accuracy of our SVM-based models to predict individual patient responses to these two chemotherapeutic drugs is comparable to our previously reported accuracy (>80%) to predict the collective responses of 273 ovarian cancer patients to 7 chemotherapeutic drugs [85].

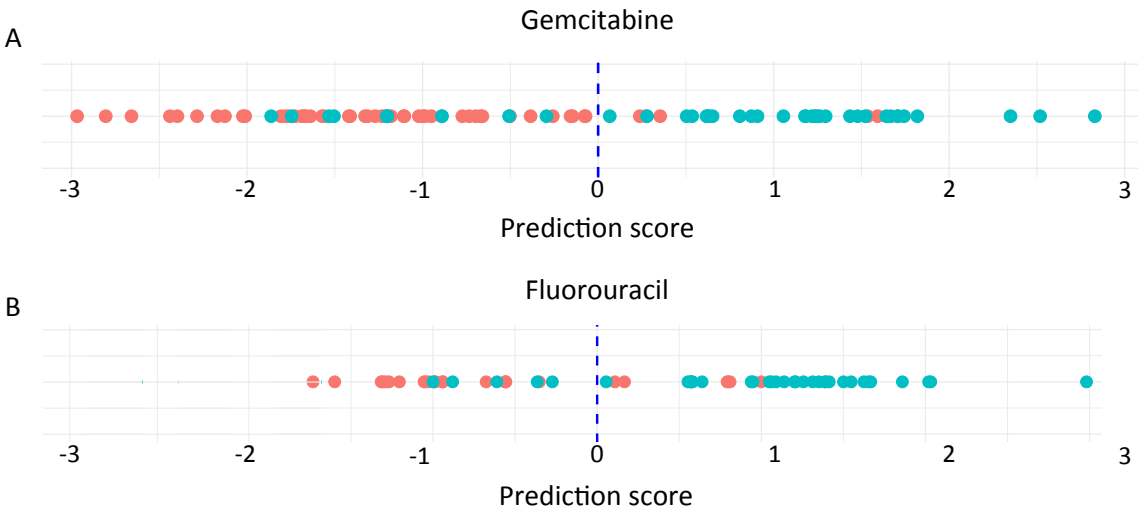


Figure 23 – Individual and aggregate prediction of response to chemotherapeutic drugs. The SVM algorithms output binary classifications for gemcitabine and 5-fluorouricil (red=drug sensitive; blue=drug resistant) established through a decision function that numerically separates tumors predicted to respond to the drug (positive score) from those predicted to be non-responders (negative score).

5.2.2 The response of individual ovarian cancer patients to standard-of-care therapies is predicted with high accuracy

The above studies generally support the potential of our SVM-RFE approach to accurately predict the drug responsiveness of individual cancer patients. To further assess the accuracy and evaluate the potential clinical usefulness of our approach, we conducted gene expression profiling of tumors collected from a randomly selected group of ovarian cancer patients and used SVM-RFE-based models to predict patient responsiveness to seven drugs often used in the treatment of ovarian cancer (carboplatin, cisplatin, paclitaxel, docetaxel, gemcitabine, doxorubicin, gefitinib).

Samples of primary tumors collected from 23 ovarian cancer patients (Table 4) were snap frozen in liquid nitrogen within one minute of surgical removal and transferred to the lab for laser capture microdissection of cancer cells and subsequent microarray gene-expression analysis (Affymetrix, U133Plus 2.0 arrays) as previously described [87]. Nearly all (21/23) of the collected samples were serous papillary ovarian cancers with the remaining two classifieds as an adenocarcinoma and a malignant mesodermal mixed tumor (MMMT). The vast majority (19/23) of the samples were derived from patients with moderate to high-grade (Grade 2-3), late stage (Stage III/IV) disease. Four of the samples were derived from patients with high-grade early stage disease (Stage I/II).

Table 4 – Clinical stage, grade and type of 23 ovarian cancer patients included in this study.

Patient ID	age at time of surgery	histopathology of tumor	stage	grade
229	58	serous papillary	IIIc	3
242	63	serous papillary	IIIb	3
272	83	adenocarcinoma	IIIb	2/3
286	52	serous papillary	IIIc	2/3
317	59	serous papillary	Ic	3
336	63	serous papillary	Ic	3
367	56	serous papillary	II	3
413	49	serous papillary	IIb	3
489	48	serous papillary	IV	3
528	66	serous papillary	IIIc	3
542	61	serous papillary	IV	3
545	74	MMMT	IIIc	3
588	71	serous papillary	IIIc	2/3
617	64	serous papillary	IIIc	2/3
620	62	serous papillary	III/IV	3
813	56	adenocarcinoma	III	1/2
992	73	serous papillary	IIIc	3
1012	75	serous papillary	IIIc	3
1122	65	serous papillary	IIIc	3
1129	65	serous papillary	IIIc	3
1145	41	serous papillary	IIIc	3
BJ1	40	serous papillary	III	3
BJ4	56	serous papillary	IIIc	3

The majority of patients (17/23) were administered chemotherapy shortly after de-bulking surgery with six patients receiving neo-adjuvant chemotherapeutic treatment prior to surgery. Most of the patients were treated with standard-of-care carboplatin/paclitaxel combination therapy (17/23). One patient was treated with carboplatin and gemcitabine, one with carboplatin and docetaxel and one with carboplatin, cisplatin and paclitaxel combination therapies. Only two patients were treated with a single drug-one with topotecan and one with doxorubicin (Table 5, Supplementary table 10).

Table 5 – Predicted and observed responses of 23 ovarian cancer patients treated with one or more of 7 chemotherapeutic drugs.

Patient	Drug	OBSERVED RESPONSE	PREDICTED Carboplatin	PREDICTED Paclitaxel	PREDICTED Cisplatin	PREDICTED Gemcitabine	PREDICTED Docetaxel	PREDICTED Doxorubicin	PREDICTED Gefitinib	PREDICTED Topotecan
229	Carbo&GEM	R (TP)	NR (FN)	NR	R	R (TP)	R	NR	R	NR
242	Carbo&Taxol	R (TP)	R (TP)	NR (FN)	NR	NR	R	R	R	NR
272	Carbo&Taxol	NR (FP)	NR (TN)	R (FP)	NR	R	R	NR	NR	NR
286	Carbo&Taxol	NR (TN)	NR (TN)	NR (TN)	NR	NR	NR	R	R	NR
317	Carbo&Taxol	R (TP)	R (TP)	R (TP)	R	R	NR	R	NR	NR
336	Carbo&Taxol	R (TP)	R (TP)	R (TP)	R	R	R	NR	NR	NR
367	Carbo&Taxol	R (TP)	R (TP)	R (TP)	NR	R	NR	NR	NR	NR
413	Carbo&Taxol	R (TP)	R (TP)	R (TP)	NR	R	R	R	NR	NR
489	Carbo&Taxol	R (TP)	R (TP)	R (TP)	NR	NR	NR	NR	NR	R
526	Carbo&Taxol	R (TP)	R (TP)	R (TP)	NR	NR	R	NR	NR	NR
542	Carbo&Taxol	R (TP)	R (TP)	R (TP)	NR	R	NR	NR	NR	NR
545	Carbo&Taxol	NR (TN)	NR (TN)	NR (TN)	NR	R	R	NR	NR	R
588	Carbo&Taxol	R (TP)	R (TP)	NR (FN)	R	R	R	NR	NR	R
617	Carbo&Taxol	R (TP)	R (TP)	R (TP)	NR	R	NR	NR	NR	NR
620	Carbo&Taxol	R (TP)	R (TP)	R (TP)	NR	R	NR	NR	NR	NR
813	Carbo/Cis/Taxol	R (TP)	NR (FN)	R (TP)	R (TP)	NR	NR	NR	NR	NR
992	Topotecan	NR(TN)	R	NR	NR	R	NR	NR	NR	NR (TN)
1012	Carbo & doxetaxel	NR (FP)	NR (TN)	NR	R	R	R (FP)	R	NR	NR
1122	Carbo & Taxol	R (TP)	NR (FN)	R (TP)	NR	R	NR	NR	NR	NR
1129	Docorubicin	R (TP)	NR	R	R	R	NR	R (TP)	NR	NR
1145	Carbo & Taxol	NR (FP)	R (FP)	NR (TN)	NR	NR	NR	NR	NR	R
BJ1	Carbo & Taxol	R (FN)	NR (FN)	NR (FN)	R	R	NR	NR	NR	NR
BJ4	Carbo&Taxol	R (TP)	NR (FN)	R (TP)	R	R	R	R	NR	NR
Totals:		16 TP,3TN,3FP,1FN								

The RNA expression profiles of significantly expressed genes were uploaded to our previously established SVM-algorithms (https://github.com/chuang95/KEA_DrugResponse) to generate drug prediction scores for each of seven chemotherapeutic drugs. We included all microarray probe sets for each gene in our analysis because, as previously demonstrated [85], the averaging of expression values over multiple probe sets can significantly reduce predictive accuracies. As described above, the predictive algorithms generate scores for each drug. Scores greater than "0" indicate a predicted positive response to the drug while scores less than "0" are predictive of drug resistance (e.g., Fig. 24A, B Fig. 35).

The majority of the 23 ovarian cancer patients analyzed were predicted to respond favorably to gemcitabine (17/23), carboplatin (13/23) and paclitaxel (13/23) (Table 5, Supplementary table 10) with less than half to cisplatin (9/23) and docetaxel (10/23). Less than third of the 23 patients were predicted to respond to doxorubicin (7/23), topotecan (4/23) or gefitinib (3/23). These predicted efficacies are generally consistent with our earlier group predictions of 273 OC patients with the exception of gemcitabine, which in our previous study, was ranked immediately behind carboplatin in predicted efficacy. This inconsistency may be attributable to sampling error due to the relatively few patients employed in the current study.

To evaluate the accuracy of our predictions, patient responses to administered chemotherapies were monitored by measurement of CA-125 values prior and subsequent to treatment according to standard criteria [88]. Patients were considered to be responsive to treatments if their respective CA-125 values dropped below normal values (<35) within 60 days of the start of chemotherapeutic treatment (Figure 24C, D, Figure 36).

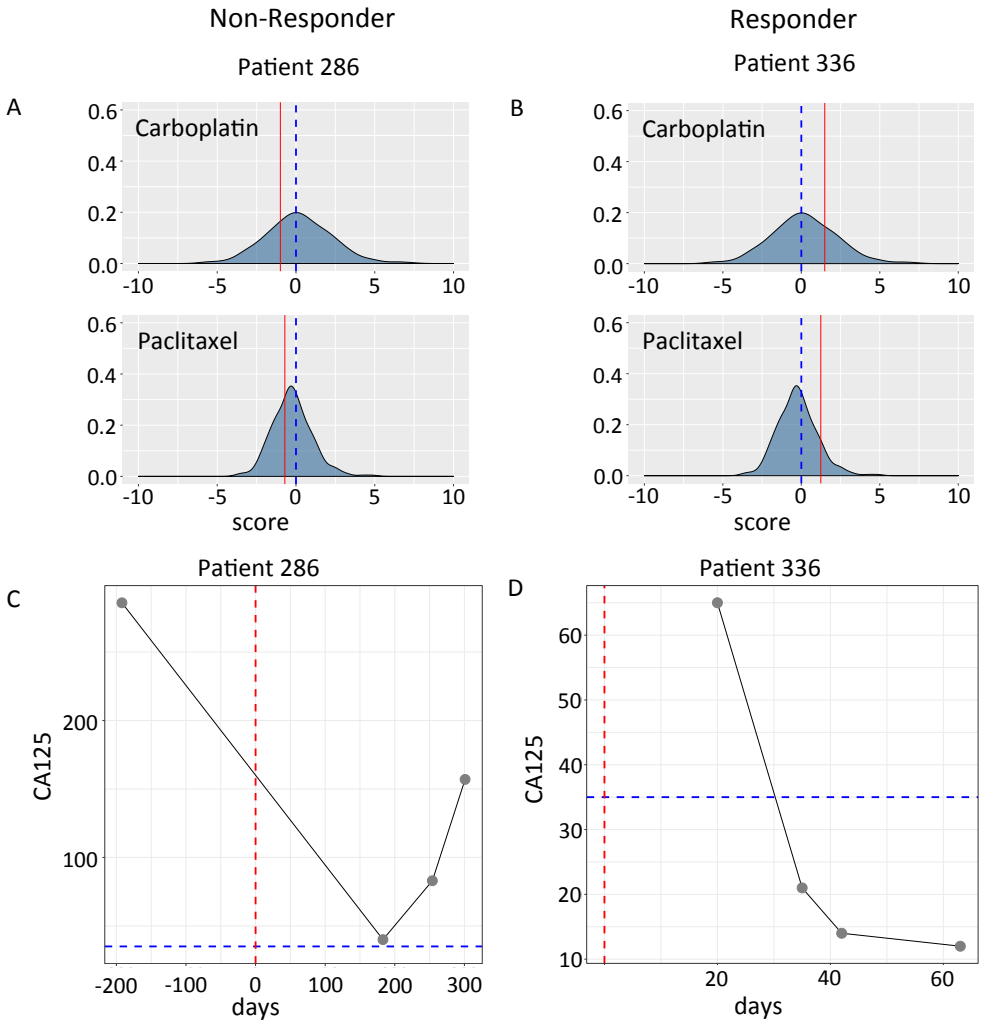


Figure 24 – Comparison of the predicted and observed responses of two ovarian cancer patients to carboplatin and paclitaxel therapies. The predicted response scores of each patient (red line) are plotted over the distribution of previously the predicted scores of 273 ovarian cancer patients [6]. Patient 286 (A) is predicted to not to respond to either drug (negative scores) while patient 336 (B) is predicted to respond to both drugs according to standard criteria [13]. Patients are considered to be responsive to treatments if their respective CA-125 values dropped below normal values (<35) within 60 days of the start of chemotherapeutic treatment (red dashed line indicates day of surgery). Patient 286 (C) is a non-responder while patient 336 (D) is a responder.

Our algorithms predict responses to individual drugs and in those few cases where patients were treated with a single drug, evaluation of the model's predictive accuracy is straightforward (e.g., patients 992 and 1129 Table 5, Supplementary table 10). However,

standard-of-care chemotherapy for ovarian cancer patients typically involves treatment with multiple drugs, most commonly, carboplatin and paclitaxel. In those cases where patients were observed to positively respond to the combination therapies, the prediction was scored as "true positive" (TP) if the patient is predicted to respond to at least one of the administered drugs (e.g., patients 317 and 588). Conversely, in cases where patients were observed to not respond to the combination therapy, the prediction was scored as "false positive" (FP) if the patient was predicted to respond to at least one of the drugs (e.g., patients 272 and 1012, Table 5, Supplementary table 10). Instances where the patient is both predicted and observed not to respond to the combination therapy are scored as "true negative" (TN) (e.g., patients 317 and 545, Table 5, Supplementary table 10) while cases where the patient responded to the combination therapy but is predicted not to respond to any of the administered drugs was scored as "false negative" (FN) (e.g., patient BJ1, Table 5, Supplementary table 10).

Based on these criteria, the computational predictions resulted in 16 TP, 3 TN, 3 FP and 2 FN. This equates to a positive predictive value (PPV) of 84.2% (sensitivity 94.1%), a negative predictive value (NPV) was 75% (specificity of 50%) equating to an overall accuracy of 83.6%. The low specificity may, in part, be due to sampling error since only six patients were observed to be non-responders in this study group.

One possible clinically useful application of our models is depicted in Figure 24. As shown (see also Figure 35), the predictive scores of an individual patient can be mapped across the distributed scores of all previously profiled patients providing information on those drugs most likely to be effective as second line treatments for an individual patient. Patient 545 was both predicted and observed (Table 5, Supplementary table 10) not to respond to carboplatin/paclitaxel treatment. An estimated 20-30% of all ovarian cancer patients treated with this standard-of-care combination therapy similarly

fail to respond to treatment [75] leaving physicians with the decision as to what to try next. ML-based models with validated high positive predictive values, such as reported here, and may provide physicians with a useful alternative to the traditional trial-and-error strategies. For example, based on the predicted responses of patient 545 to the possible second line drugs modeled in this study, gemcitabine stands out as a preferred choice.

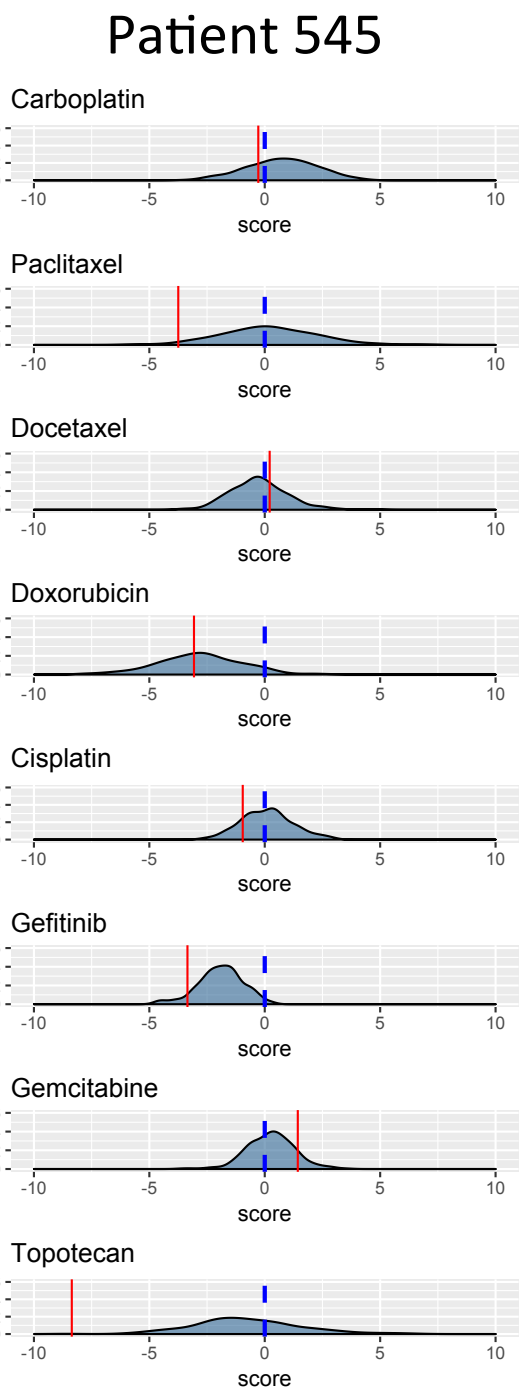


Figure 25 – Algorithms with high positive predictive value (PPV) may be of particular clinical benefit in the selection of alternative second-line chemotherapies. Patient 545 was predicted (and observed, see Table 5) not to respond to standard-of-care carboplatin/paxitaxel therapy. Of possible second-line therapies, gemcitabine is predicted to be the preferred choice.

5.3 Discussion

Cancer is a complex disease. The fact that there are a multitude of possible molecular paths to developing even the same type of cancer explains, in large measure, why the response to any given chemotherapeutic drug can be highly variable across patients [89]. Our increasing ability to accurately profile individual patient tumors on the molecular level is widely viewed as a promising resolution to this problem. Indeed, a major goal of modern cancer medicine is the ability to accurately predict optimal drug therapies based upon the personalized molecular profiles of individual patient tumors.

Accurate predictions in cancer biology, as in all areas of science, can be based upon established cause-and-effect relationships or upon highly significant correlations detected in large sets of relevant data. While we are well on our way to the day when we may fully understand the molecular causes of all cancers and treat them accordingly, we are not there yet. One promising interim solution is the application of prediction algorithms derived from ML-detected correlations between the molecular profiles of large numbers of cancers and associated responses to variety of therapeutic drugs [90].

We recently reported on the use of our open access SVM-based algorithms to accurately (>80%) predict the collective response of 273 ovarian cancer patients to seven commonly prescribed chemotherapeutic drugs [85]. In this study, we were interested in evaluating the performance of our approach to predict individual patient responses to drugs based on gene expression profiles of each individual tumor. Employing gene expression (RNAseq) profiles of 152 cancer patients downloaded from the TCGA

database, we were able to predict the response of individual patients treated with either gemcitabine or 5-FU with >81% accuracy. In a second study, the response of individual ovarian cancer patients to seven commonly prescribed chemotherapeutic drugs, based on microarray gene expression profiles of each patient's tumor, was predicted with an overall accuracy of 83% and a PPV of 84%. The high PPV of our algorithms across multiple drugs suggests a potential clinical utility of our approach to identify promising second-line treatments for patients failing standard-of-care first-line therapies.

It should be noted that although our models have, thus far, focused on the predicted response of cancer patients to current standard-of-care drug therapies for which sufficient datasets are available, the approach is equally as well applicable to emerging immuno- and other targeted gene therapies where patient responses are also known to be variable and likely dependent upon the personalized genomic makeup of individual tumors [91].

CHAPTER 6. FUTURE WORK AND CONCLUSION

Overall this work in totality reflects our deep seeded belief that algorithms and heuristics that advance precision medicine should be open source to allow the maximum benefit for humanity. Through two major types of analysis my present work has moved the needle forward in terms of genomic comparisons of sequences useful for comparative genomics (i.e. Boolean based genomic topology) as well as key formatting of data types to allow for automated and rapid analysis that scales with arbitrarily large datasets.

This technique enables the comparison of highly divergent genomic sequences and we utilize these heuristics to perform hPCA clustering of bacterial and viral sequences (Fig. 26A). As well as clustering based on pairwise distance of Boolean analysis shown in Figure 26B. In addition to finding a number of known relationships between bacteria and viruses, we also found evidence of previously cryptic relationships between these bacterial and viral genomes (Appendix A1, Table 6, 7). These relationships highly suggestive of shared genomic content through horizontal gene transfer, and provide a new perspective to examine these putative new relationships between of bacteria and viruses. We prepared a k-merized data set of 2569 bacteria and 1754 virus sequences for further study. Future work will allow clustering of arbitrarily large sets of genomes across diverse phyla, something that is not possible with conventional alignment based heuristics. We also see the possibility of capturing all deposited sequences for a given outbreak and plot newly sequenced isolates in the context of the complete data, again, something that is not feasible using older frameworks.

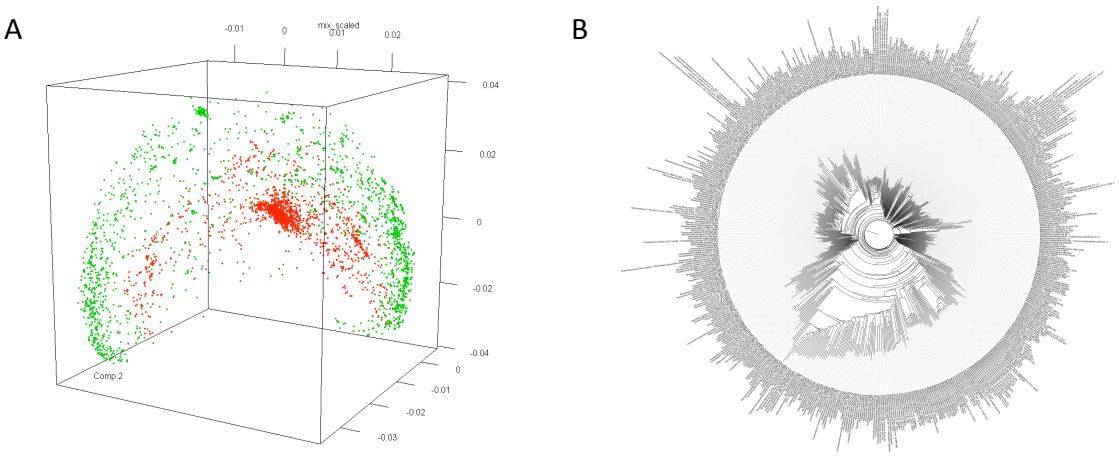


Figure 26 – Scaled Linear algebra and Boolean analysis. A) Orthogonal transformation of the matrices from 2569 number bacteria (green) and 1754 number viruses (in red). B) Hierarchical tree based on pairwise distance of Boolean analysis.

After achieving scaled clustering of microbial genomic data (as seen above), we went ahead and extended our drug response prediction algorithm to analyze a larger set of cancer drugs. In Chapter IV and V, we analyzed 7 FDA proved drugs for ovarian cancer patients. Since we didn't want to limit our algorithm only on prediction drug response for ovarian cancer patients, we selected 95 FDA-approved drugs that were screened with the NCI-60 cell lines and that have GI-50/IC-50 data. We built a model for each drug, and initially test these models on ovarian cancer patient clinical trials (GSE20565, GSE18521, GSE30161). As shown in Figure 27, the density plot of drug response for two example drugs, Thiotepa and Vorinostat, are similar to the predicted efficacy result for Carboplatin. Based on these results, we can start to predict which drug could be effective for treating ovarian cancer patients. Looking into the literature and without our prior knowledge, it turns out that both Thiotepa and Vorinostat have recently been utilized in clinical trials for ovarian cancer [92, 93]. Looking at global scale response off of our predication model could be a way in which to help prioritize specific drug candidates above others, and this type of 'in silico trial screening' will only improve as our drug

prediction models improve. Next steps include being able to carry our prospective clinical trials to match drugs with patient profiles.

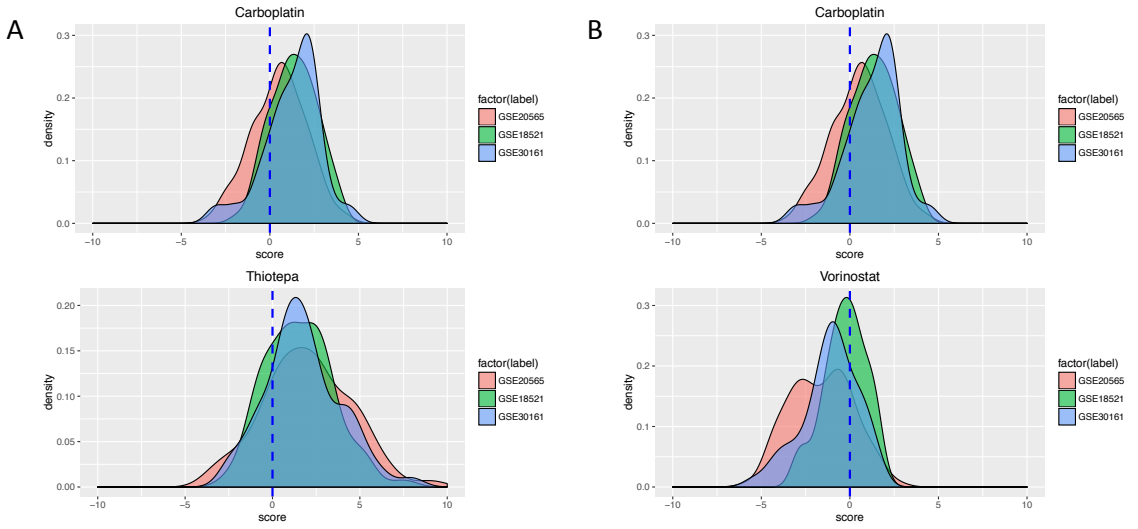


Figure 27 – Density plot of aggregate prediction scores for 3 GEO data sets of 273 ovarian cancer patients and the predicted group response rate for drug A) Thiotepa and B) Vorinostat comaparing to drug Carboplatin.

In addition to the work above we are developing the project's website to allow interested parties to view and download the source code, as well as obtain additional information about the algorithmic basis of our current version (http://vannberg.biology.gatech.edu/data/DrugResponse/). We envision that this work, alongside other academic and commercial groups, forms the basis of a broad reaching truly open source drug sensitivity algorithm development challenge moving forward. For this, developers can access large sets of pre-formatted data and the competition will be set up with an initial working pipeline for computer engineers to be able to start and to then subsequently optimize to continue to improve and optimize the performance of prediction. To date our group has achieved data wrangling of nearly all of the available NCBI GEO datasets into one matrix for a given array. For example, we have processed data from the Affymetrix U133 Plus 2.0 (GPL570) array which will include 124,139

separate arrays processed into a single matrix to allow for direct normalization to the analysis set of interest (in this case the NCI-60 set). Although we have utilized MAS5 normalization for the work outlined in this thesis future work will utilize another normalization technique more precisely suited for probe level analysis. Also, instead of gene level RNA-seq data, a model built on exome level RNA-seq data could improve the performance of drug response prediction. These approaches may increase the accuracy, sensitivity and specificity of our analysis. By combining all of this data together, we provide a matrix format that is inherently easy to access and manipulate. By doing so we can compare any new cancer sample with all others to allow for algorithmic prediction of not only drug response, but also putative tissue of origin, clonal heterogeneity, stromal cell contamination, immune cell infiltration, and other key metrics that will be useful in future therapeutic decision making. By making this resource open and easy to use we can envision how cancer care will progress and how these advancements can be added into future electronic health care record (EHR) fields. Our lab has also attempted to automate plots for the assessment of drug response (i.e., CA125 plots) and more work will need to be done on this to ultimately automate and integrate all data which can help to determine drug response into a cohesive analysis. We invasion that neural networks will serve to benefit key modules of our work flow, and might useful to improve the performance of current algorithm. By minutely understanding the exact tumor response for each drug and dose we can continue the virtuous circle of continued training of the drug response algorithms to further increase accuracy. All of these developments add transparency and allow developers to learn from each other in real time and to iterate to improve these algorithms and we look forward to opening up a drug sensitivity prediction challenge in the near future to allow academic and commercial researchers locally and abroad to compete to create better and better predictions.

Through this work I've been able to produce open source scripts that involve both supervised as well as unsupervised machine learning, and my work serves as an initial template to power collaborative efforts in topics such as drug response for cancer drugs, outbreak discovery for infectious diseases, and other areas not covered in this dissertation including variant calling for drug resistance and other important topics.

Additional optimizations and data scaling remain to be carried out, but this dissertation presents a defined answer to the numerous studies in this space that shared very little true open source code when assessing drug response. I believe that clarity of thought is reflected in the elegance of a given algorithm and data format, and throughout this dissertation implementation of the *kmatrix* format, alongside the Boolean *XOR* function form what must be the absolute most succinct and elegant approach to this given problem. In terms of machine learning those that are at the forefront of the field have suggested a similar philosophical thought, and I hope that through this dissertation others will also search out this type of philosophical aim to produce algorithms that are not only functional, but also elegant.

APPENDIX

A.1 Relationships between bacteria and viruses

This technique enables the comparison of highly divergent genomic sequences and we utilize these heuristics to perform clustering of bacterial and viral sequences (Fig 26, Table 6, 7). From genomic topology analysis, we find examples that make sense in terms of likely horizontal gene transfer such as a close link between *Sulfolobus* virus *STSV1* and *Sulfolobus islandicus M*; *Sulfolobus* virus *STSV1* and Sulfolobus solfataricus; *Sulfolobus* virus *STSV1* and *Sulfolobus acidocaldarius*; *Sulfolobus* virus *STSV1* and *Sulfolobus tokodaii* [94]; *Salmonella* phage *ST64B* and *Escherichia coli ED1a* [95]; *Klebsiella* phage *phiKO2* and *Escherichia coli O26* [96]. However, this analysis also revealed several unexpected relationships between *Klebsiella* phage *phiKO2* and *Enterobacter cloacae EcWSU1*; *Enterobacteria* phage *N15* and *Pectobacterium atrosepticum SCRI1043*, among others.

We demonstrate that our approach can analyze an ancient genome such as *Acidianus* bottle shaped virus, which infect archaea [97], in the context of other viruses. By using grep function, we output pairwise relationship between *Acidianus* bottle shaped virus and all other species in our database, then we sorted by the hPCA distance. We find out close virus spices have stronger relationship than other random picked and bacteria (Fig. 28). Also, we can to study the relationship between this virus and other bacteria. So, we show a circus plot of several bacteria, which have low hPCA distance with the virus, and five random picked *Methylobacterium* family bacteria (Fig. 28). We can see the

Methylobacterium family are far away from our virus, because they are less related than the *Vulcanisaeta moutnovskia* [98] to the virus, and it is an Archaea, which might be infected by *Acidianus* bottle shaped virus. Besides, *Caldivirga maquilingensis*, *Ignisphaera aggregans*, *Vulcanisaeta* distribute, *Desulfurococcus fermentans*, are all Archaea. This suggests that there is relationship between virus and infected bacteria on genome perspective. For another example, we can study on an unclassified sequence, and suggest classification. We pick an unclassified virus, *Pyrococcus abyssi* virus 1. By using grep function, we output relationship between this virus and all other species in our database. After sorting the output by hPCA distance, we study the top 12 species (log-transformed hPCA distance higher than 2.5) and their lineage (Table 6). From the table, we can see that most of these known species are belong to Viruses; dsDNA viruses, no RNA stage; *Caudovirales* family. So, we can positively suggest that *Pyrococcus abyssi* virus 1 can be classified to viruses; dsDNA viruses, no RNA stage; *Caudovirales*.

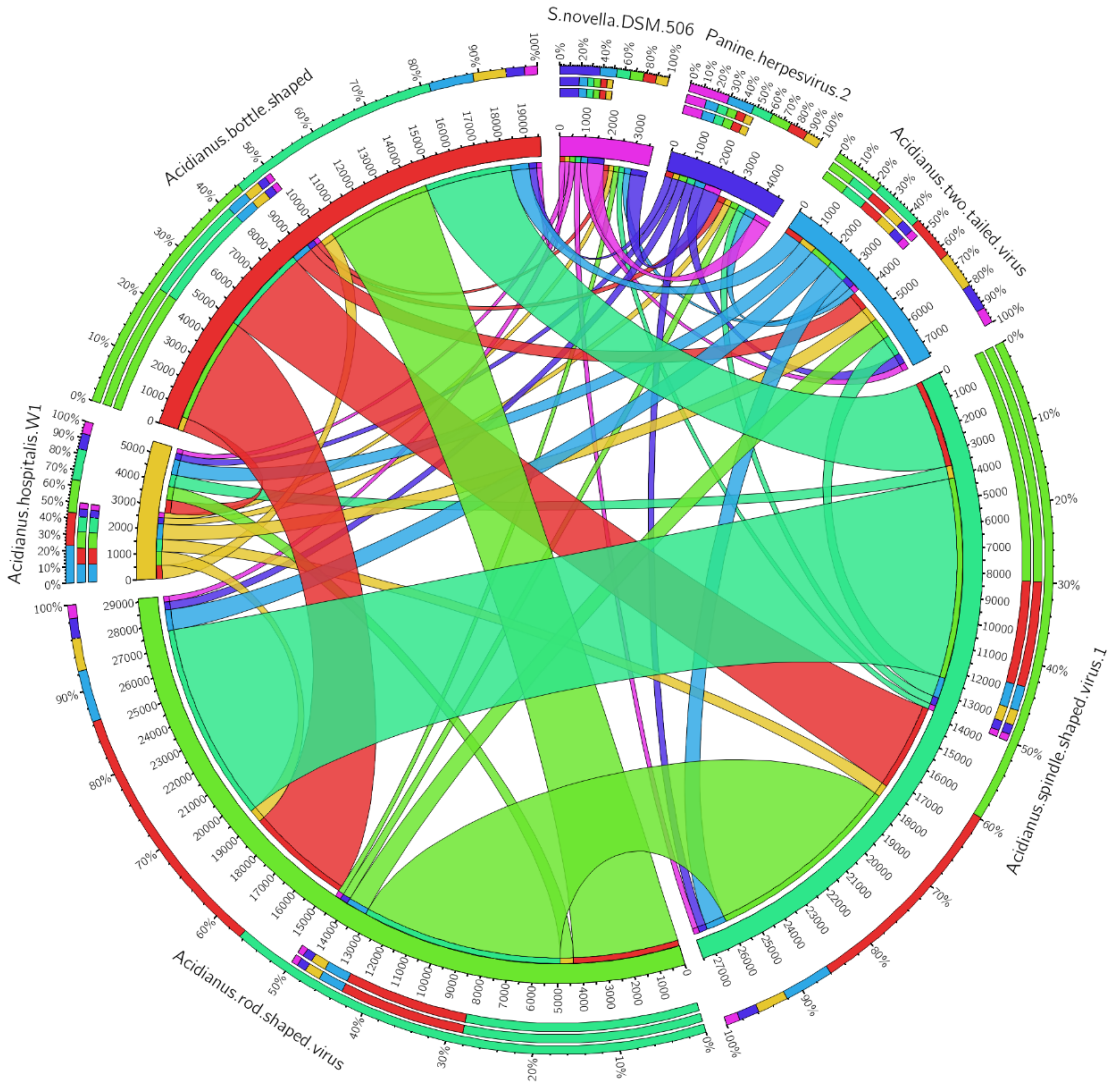


Figure 28 – Circus plot of five Acidianus family virus, one random herpesvirus and one random Starkeya.novella bacteria. The pairwise relationship are present by 10/(hPCA distance), so the thicker the connect line is, the closer they are.

Combining our Boolean Analysis algorithm with hPCA result can be more advanced to study certain species. For example, we pick another unclassified virus, *Sulfolobus tengchongensis* spindle-shaped virus 1 (*STSV1*). We output the pairwise relationship between *STSV1* and other species in our database and we sort the output by the average of log transformed hPCA and Boolean Analysis distance. The top 10 bacteria

species are belonging to *Sulfolobus islandicus* family (Table 7). It's positive to believe that *STSV1* is a virus that infects *Sulfolobus islandicus* family.

Table 6 – Pairwise relationship between *Pyrococcus abyssi* virus 1 and top 12 close species, and their lineage.

S1	S2	hPC	-log	Lineage
Xanthomonas.phage.OP2	Pyrococcus.abyssi.virus.1	0.00113	2.946938	Viruses; dsDNA viruses, no RNA
Pyrobaculum.spherical.virus	Pyrococcus.abyssi.virus.1	0.001583	2.800532	Viruses; dsDNA viruses, no RNA
Morganella.phage.MmP1	Pyrococcus.abyssi.virus.1	0.002079	2.682144	Viruses; dsDNA viruses, no RNA
Hyperthermophilic.Archaeal.Virus.1	Pyrococcus.abyssi.virus.1	0.002292	2.639807	Viruses; dsDNA viruses, no RNA
Phormidium.phage.Pf.WMP3	Pyrococcus.abyssi.virus.1	0.00238	2.623421	Viruses; dsDNA viruses, no RNA
Burkholderia.phage.KS9	Pyrococcus.abyssi.virus.1	0.002698	2.569013	Viruses; dsDNA viruses, no RNA
Yersinia.pestis.phage.phiA1122	Pyrococcus.abyssi.virus.1	0.002732	2.563507	Viruses; dsDNA viruses, no RNA
Enterobacteria.phage.SSL.2009a	Pyrococcus.abyssi.virus.1	0.002814	2.550622	Viruses; dsDNA viruses, no RNA
Enterobacteria.phage.T7	Pyrococcus.abyssi.virus.1	0.00284	2.546674	Viruses; dsDNA viruses, no RNA
Burkholderia.phage.phiE255.	Pyrococcus.abyssi.virus.1	0.002842	2.546378	Viruses; dsDNA viruses, no RNA
Enterobacteria.phage.13a	Pyrococcus.abyssi.virus.1	0.002894	2.538576	Viruses; dsDNA viruses, no RNA
Burkholderia.phage.Bcep176	Pyrococcus.abyssi.virus.1	0.003103	2.508226	Viruses; dsDNA viruses, no RNA

Table 7 – Pairwise relationship between *Sulfolobus tengchongensis* spindle-shaped virus 1 and top 10 close Bacteria sorted by the average of log transformed hPCA and Boolean Analysis distance.

S1	S2	hPCA	$\Delta\Gamma_k$	Average
Sulfolobus.virus.STSV1	Sulfolobus.islandicus.LAL14.1	3.3619	0.1783	1.7701
Sulfolobus.virus.STSV1	Sulfolobus.islandicus.HVE10.4	3.3430	0.1728	1.7579
Sulfolobus.virus.STSV1	Sulfolobus.islandicus.M.14.25	3.2903	0.1733	1.7318
Sulfolobus.virus.STSV1	Sulfolobus.islandicus.Y.G.57.14	3.2702	0.1714	1.7208
Sulfolobus.virus.STSV1	Sulfolobus.islandicus.Y.N.15.51	3.2490	0.1709	1.7100
Sulfolobus.virus.STSV1	Sulfolobus.islandicus.REY15A	3.2314	0.1786	1.7050
Sulfolobus.virus.STSV1	Sulfolobus.islandicus.L.S.2.15	3.1989	0.1688	1.6839
Sulfolobus.virus.STSV1	Sulfolobus.islandicus.L.D.8.5	3.1872	0.1679	1.6775
Sulfolobus.virus.STSV1	Sulfolobus.islandicus.M.16.27	3.1653	0.1718	1.6686
Sulfolobus.virus.STSV1	Sulfolobus.islandicus.M.16.4	3.1355	0.1724	1.6539

A.2 Feature selection algorithms comparison

Take drug Carboplatin as example, we compare our approach to two common used feature selection techniques, PCA and sequential forward selection.

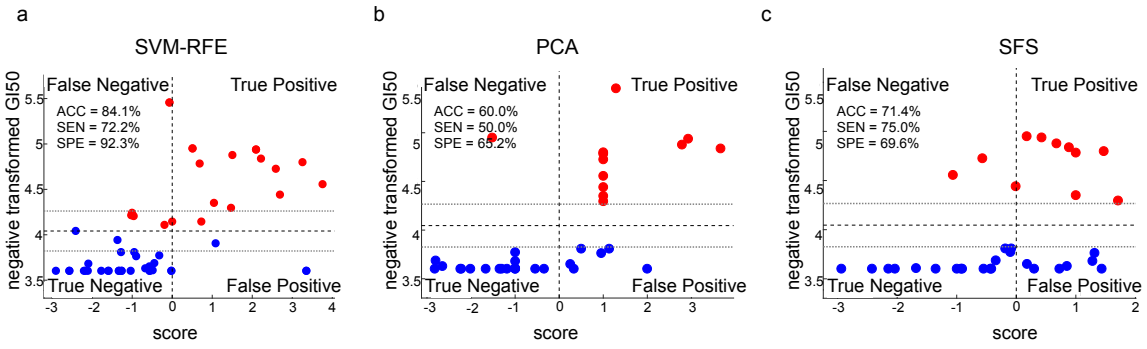


Figure 29 – Comparing LOOCV evaluation of three feature selection approaches. The statistical report (accuracy, sensitivity and specificity) in the figure shows our SVM-RFE yielded best result among these techniques.

A.3 Drug response prediction for all 6 drugs

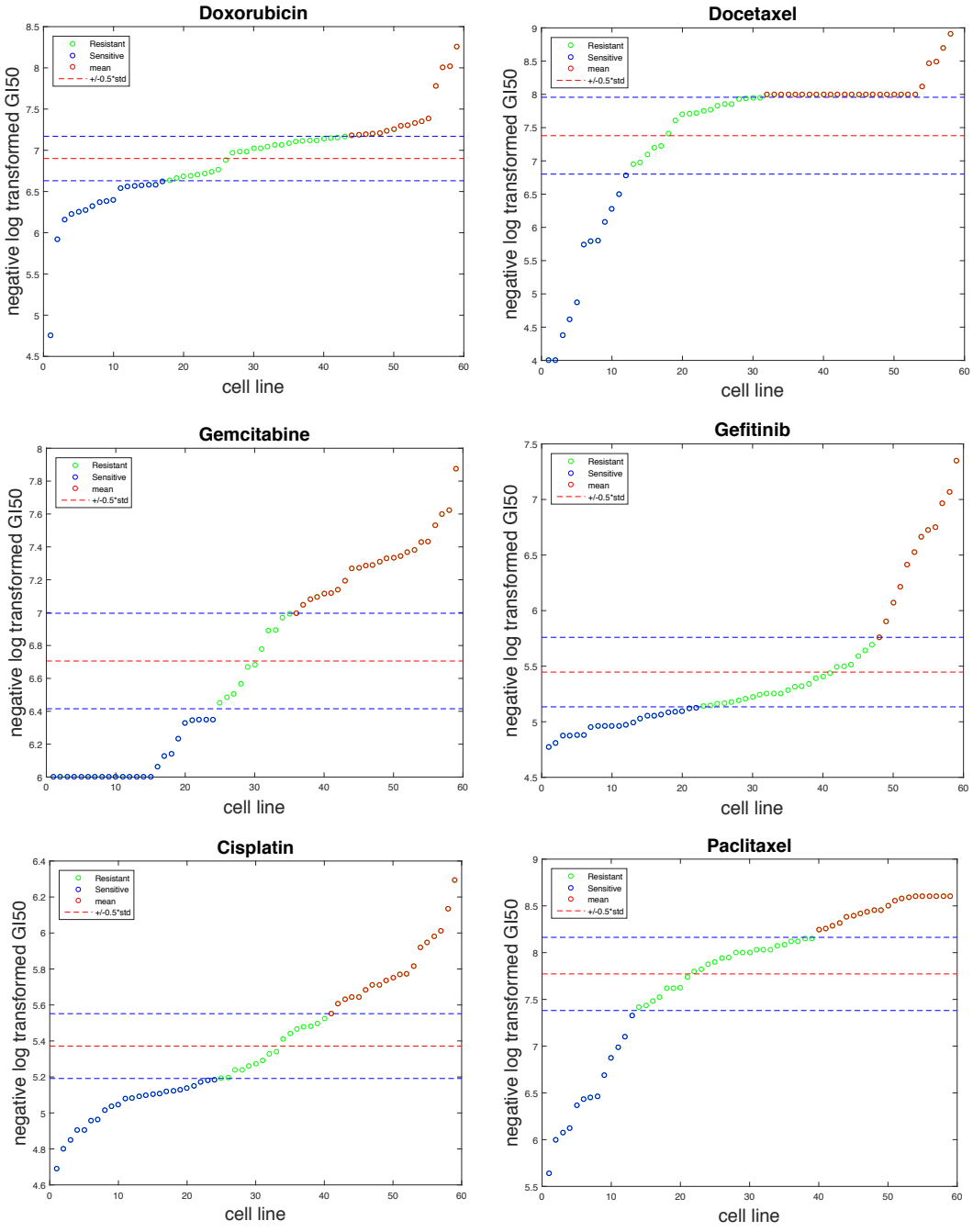


Figure 30 – Labels of response to each drug of NCI-60 cell lines are determined by IC50 value. Here we show negative log transformed IC50 value of NCI-60 cell lines for each drug. The higher the negative log transformed 50 value, the more sensitive response cell lines are. Data between mean+0.5*std and mean-0.5*std are removed

due to undetermined of response. Data points beyond mean+0.5*std are labeled as 1(response) and data points below mean-0.5*std are labeled as 0(not response).

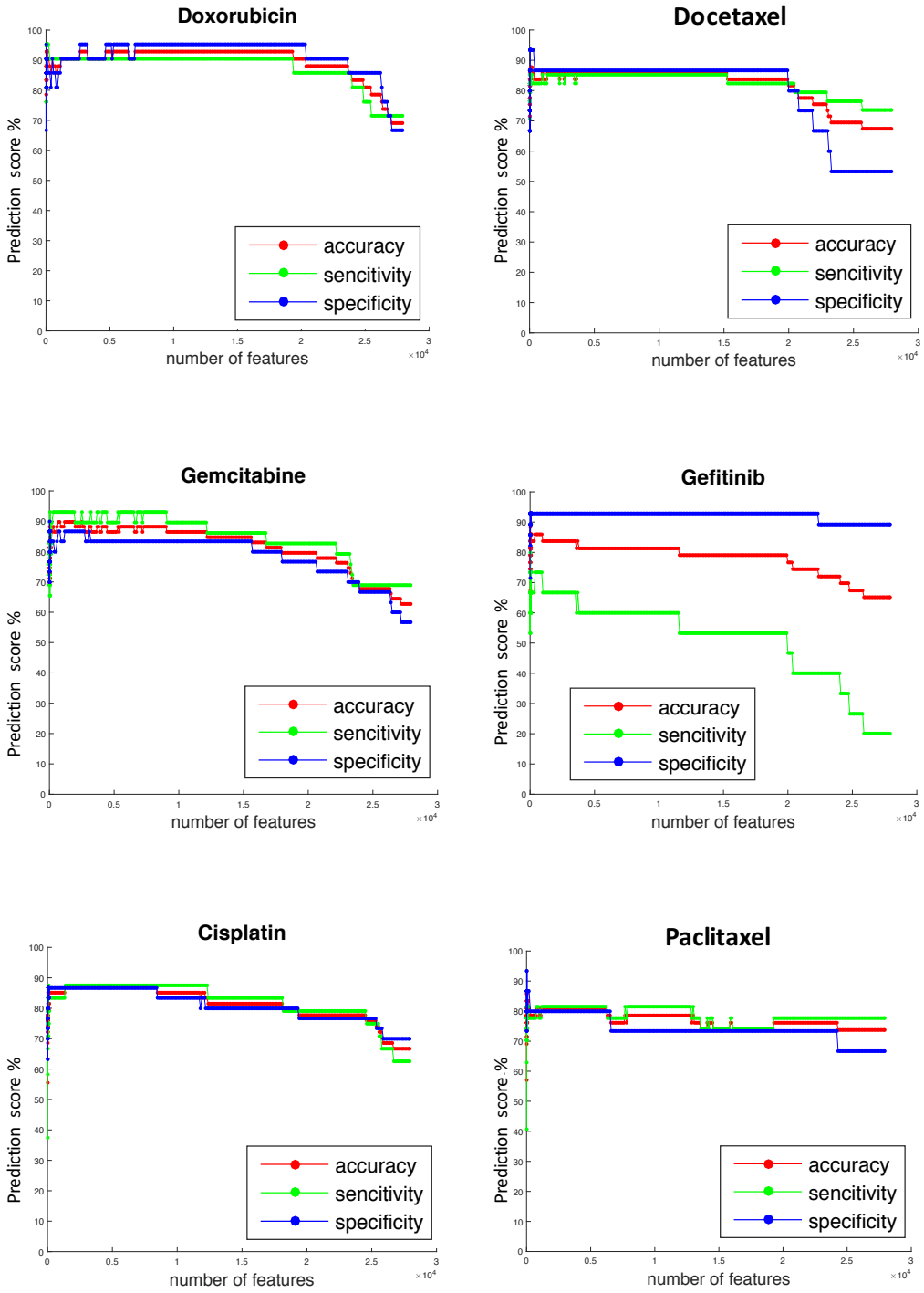


Figure 31 – RFE performance for each drug. We plot accuracy, sensitivity, and specificity for each step of RFE.

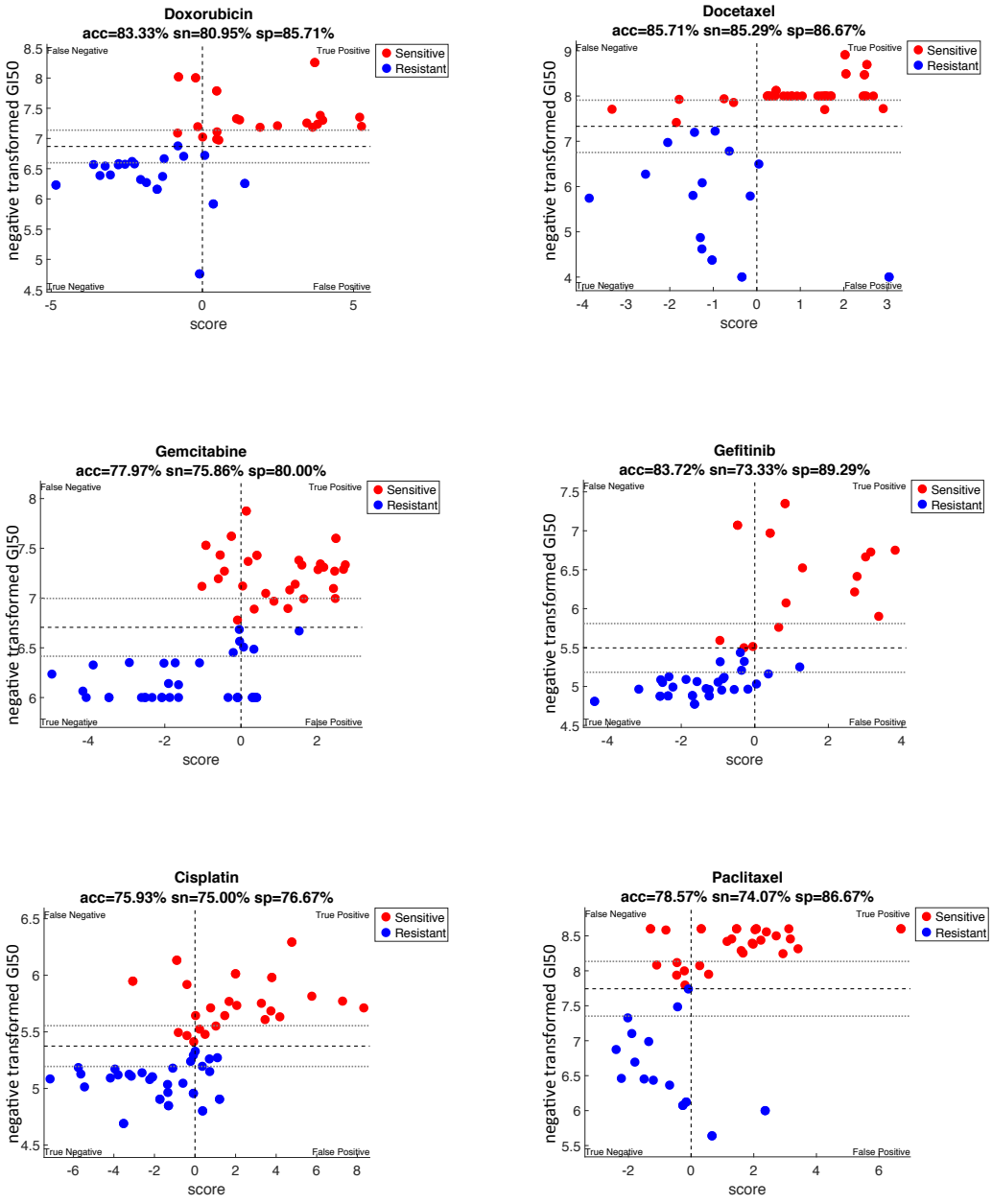


Figure 32 – Leave-one-out cross-validation for each model. We plot GI-50 of each data points (y axis) against its prediction score (x axis).

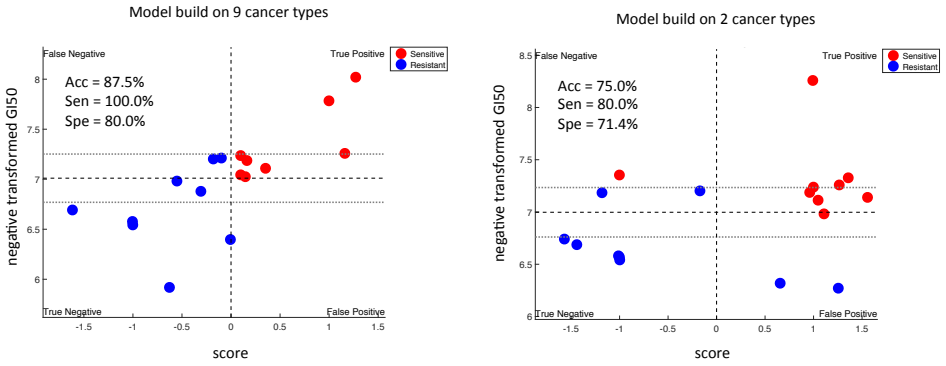


Figure 33 – We first built a model using 18 cell lines from lung cancer and melanoma (i.e. 2 cancer types) and another model using 18 cell lines from brain, breast, lung, leukemia, renal, colon, ovarian, prostate and melanoma cancer cells (i.e. 9 cancer types). We LOOCVs for two models and show the first model performs better.

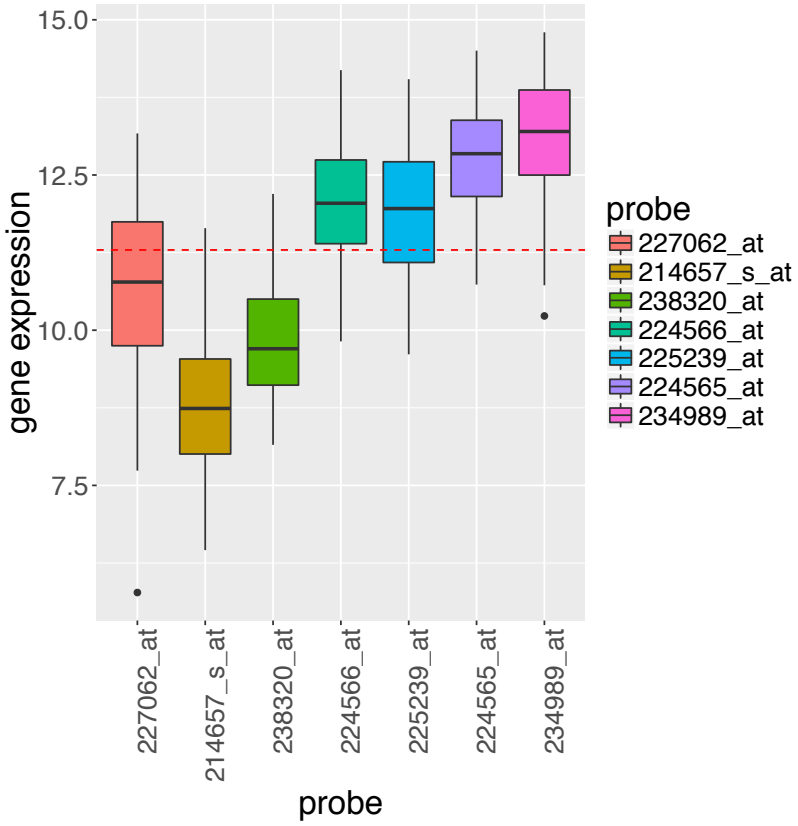
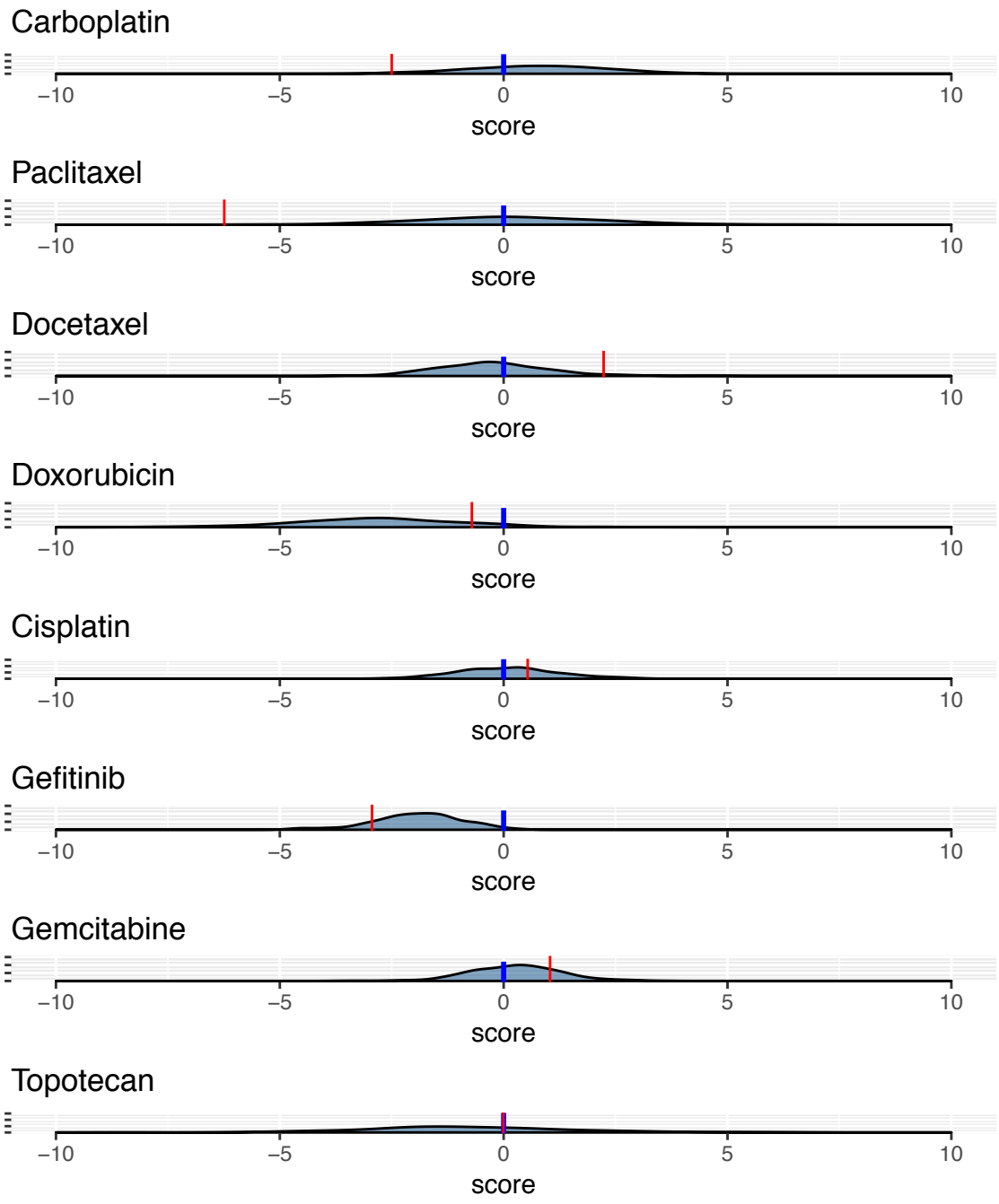
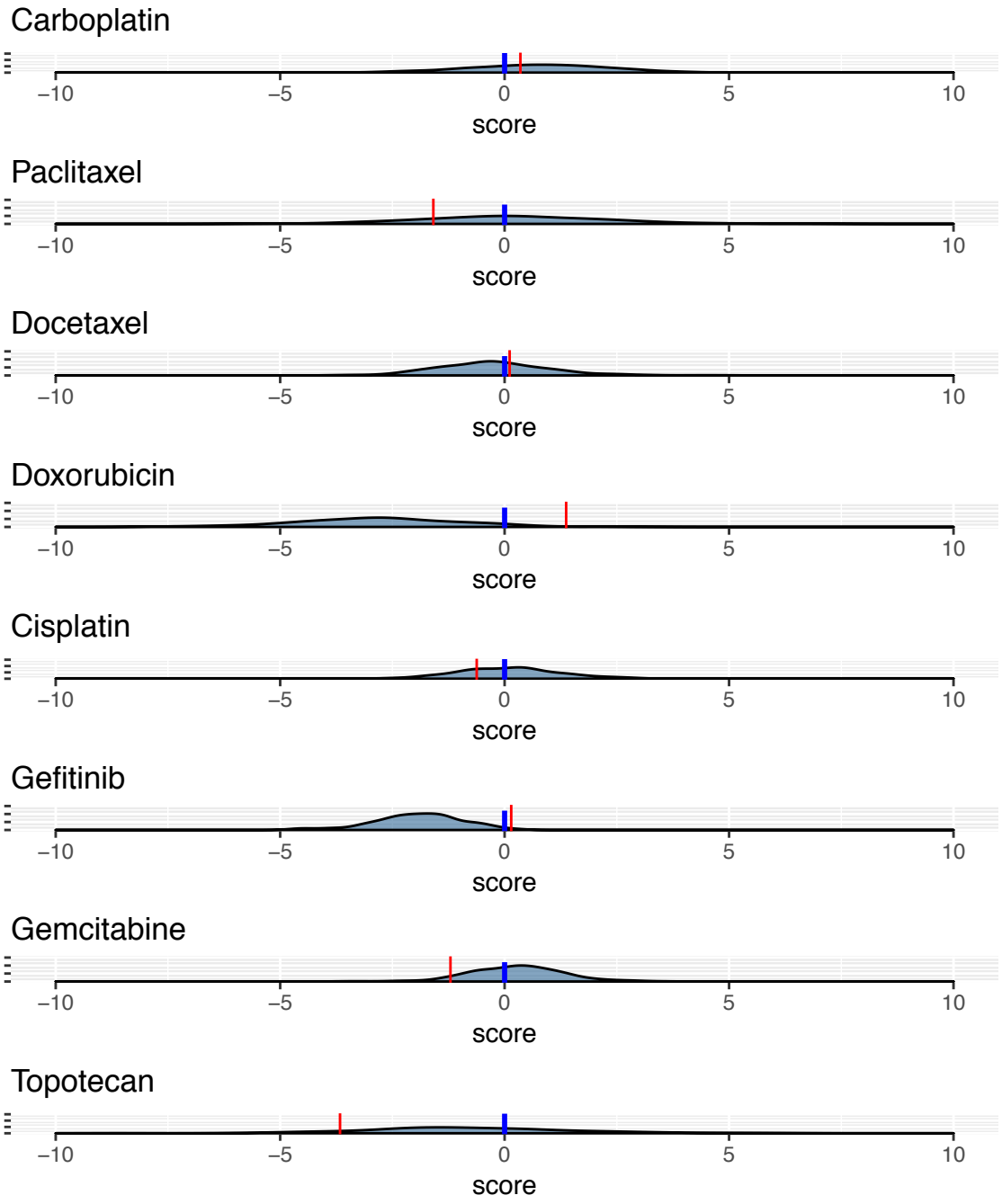


Figure 34 – We select one gene (MIR612_///_NEAT1), which has 7 probe sets in our NCI-60 gene expression data. This box plot shows the expression variation for probes in one gene. The red dot line shows the average (11.3) of all probes to gene level.

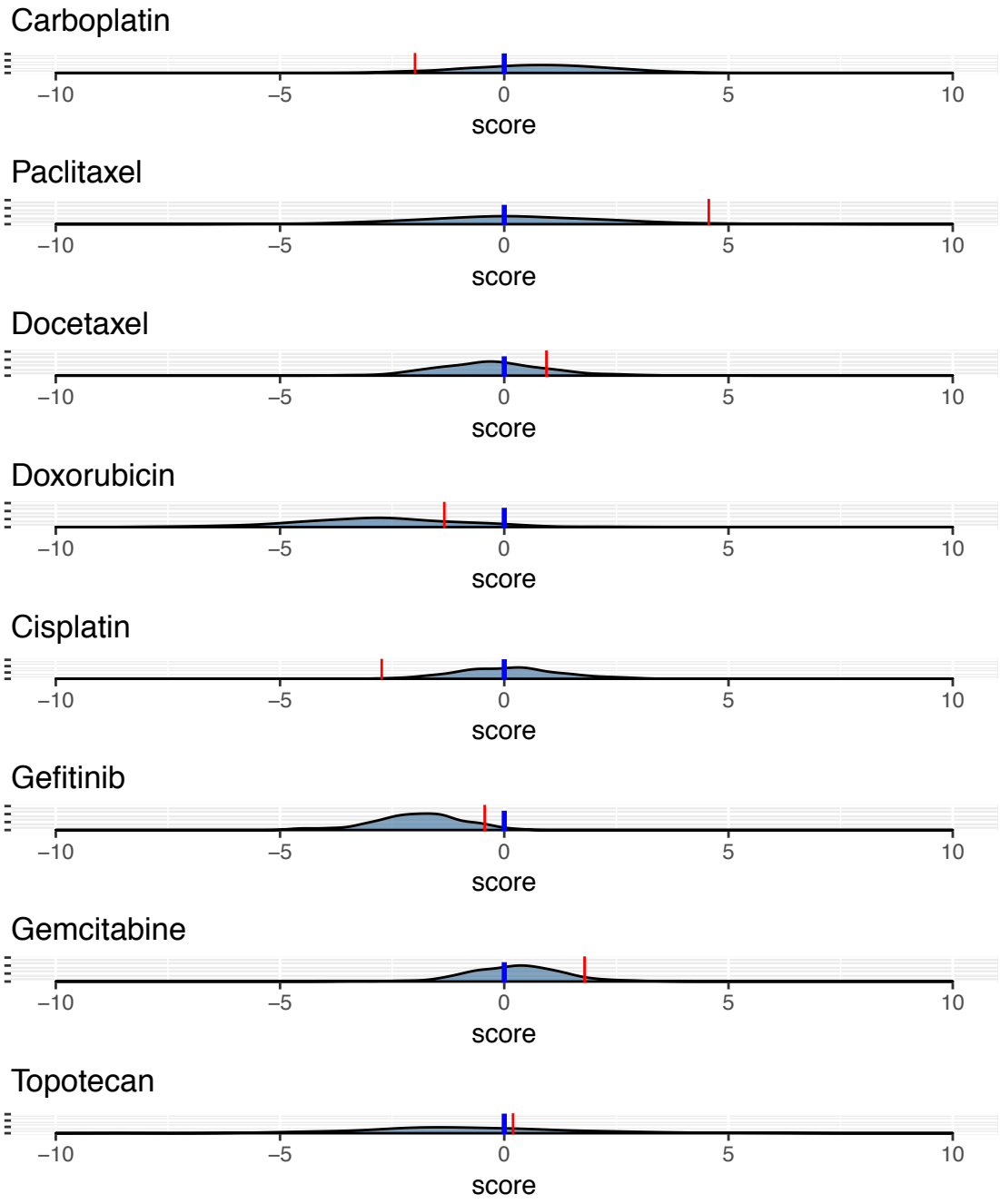
Patient 229


Carboplatin
−10
−5
0
5
10
score
Paclitaxel
−10
−5
0
5
10
score
Docetaxel
−10
−5
0
5
10
score
Doxorubicin
−10
−5
0
5
10
score
Cisplatin
−10
−5
0
5
10
score
Gefitinib
−10
−5
0
5
10
score
Gemcitabine
−10
−5
0
5
10
score
Topotecan
−10
−5
0
5
10
score

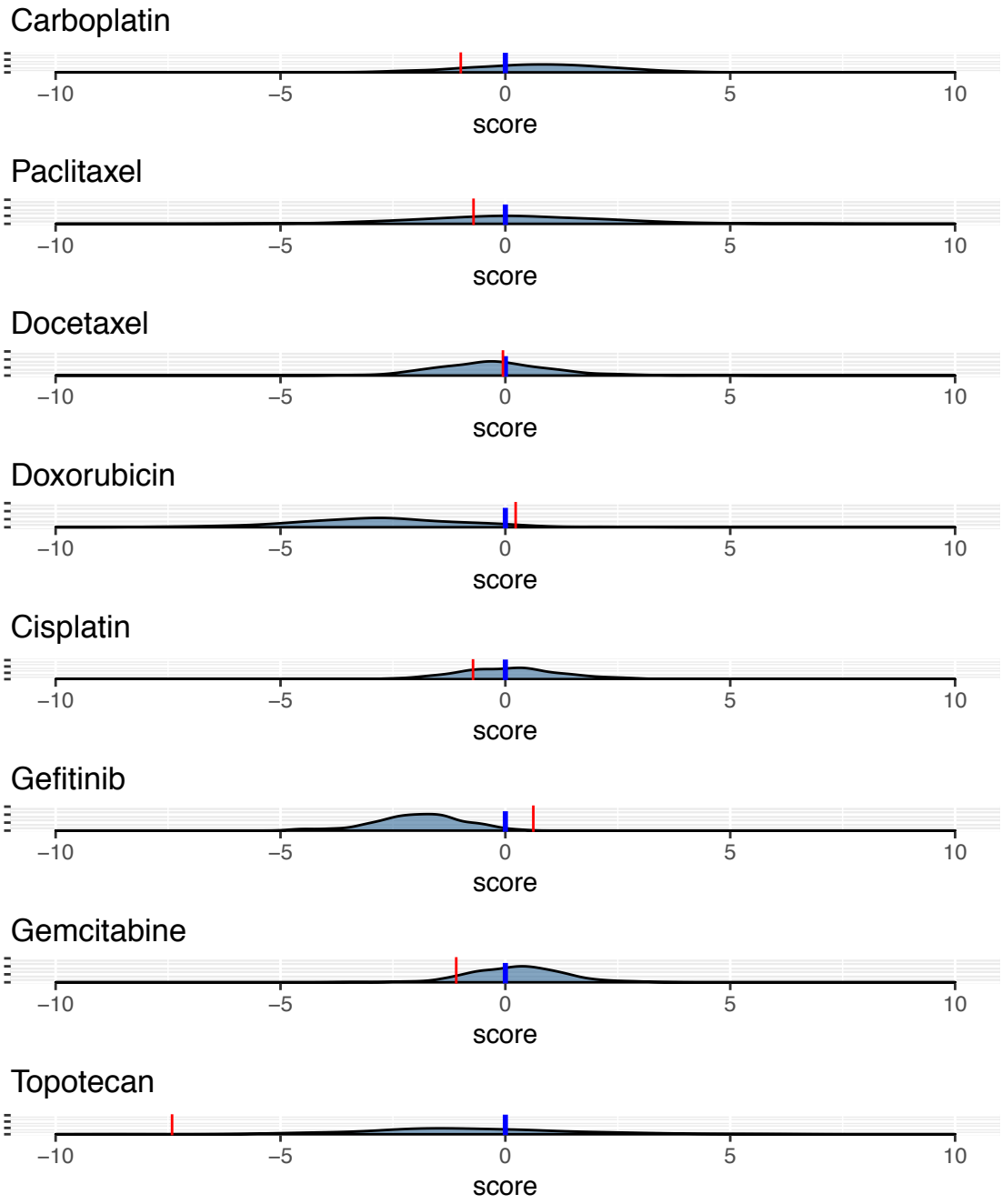
Patient 242


Carboplatin
−10
−5
0
5
10
score
Paclitaxel
−10
−5
0
5
10
score
Docetaxel
−10
−5
0
5
10
score
Doxorubicin
−10
−5
0
5
10
score
Cisplatin
−10
−5
0
5
10
score
Gefitinib
−10
−5
0
5
10
score
Gemcitabine
−10
−5
0
5
10
score
Topotecan
−10
−5
0
5
10
score

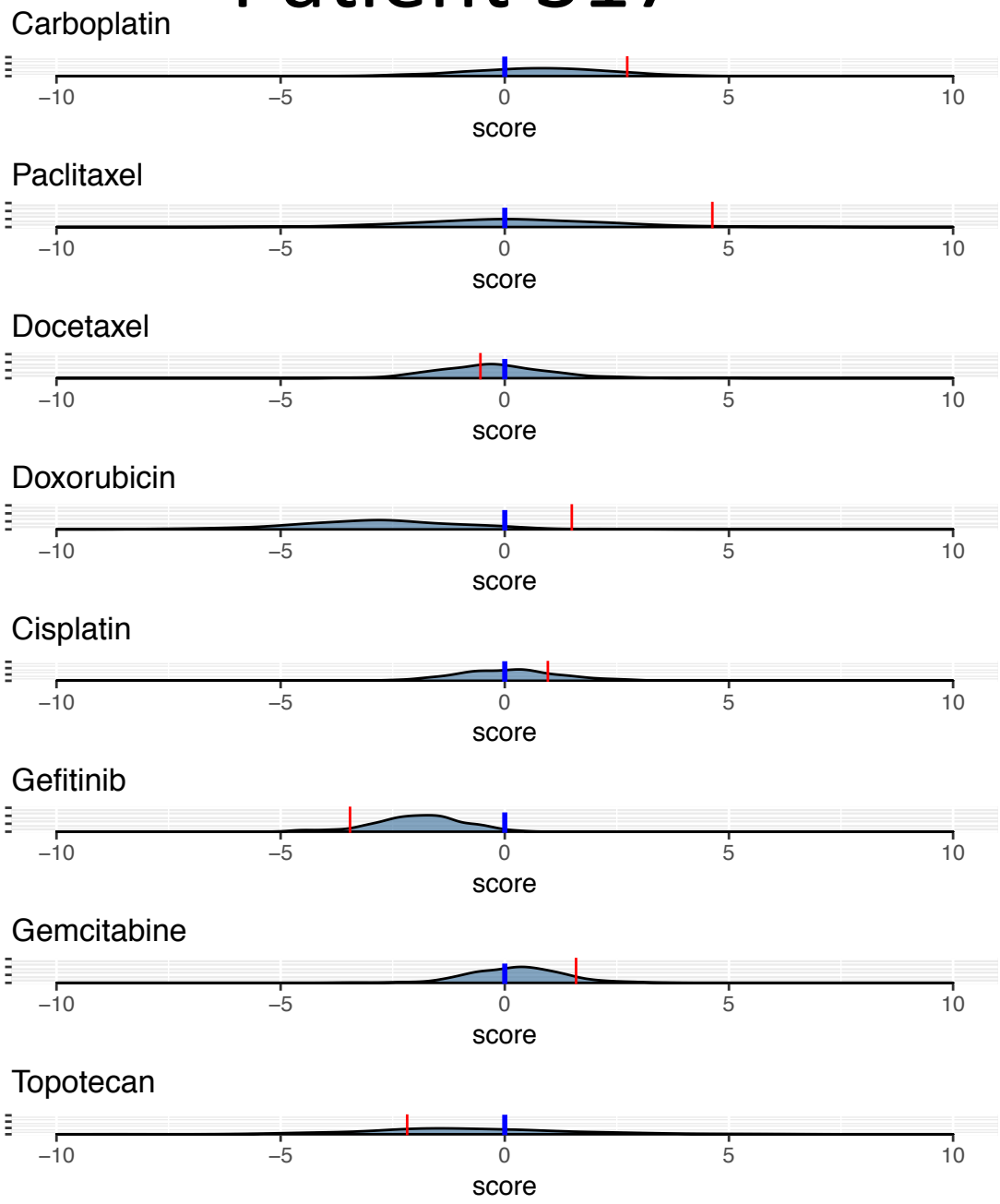
Patient 272


Carboplatin
−10
−5
0
5
10
score
Paclitaxel
−10
−5
0
5
10
score
Docetaxel
−10
−5
0
5
10
score
Doxorubicin
−10
−5
0
5
10
score
Cisplatin
−10
−5
0
5
10
score
Gefitinib
−10
−5
0
5
10
score
Gemcitabine
−10
−5
0
5
10
score
Topotecan
−10
−5
0
5
10
score

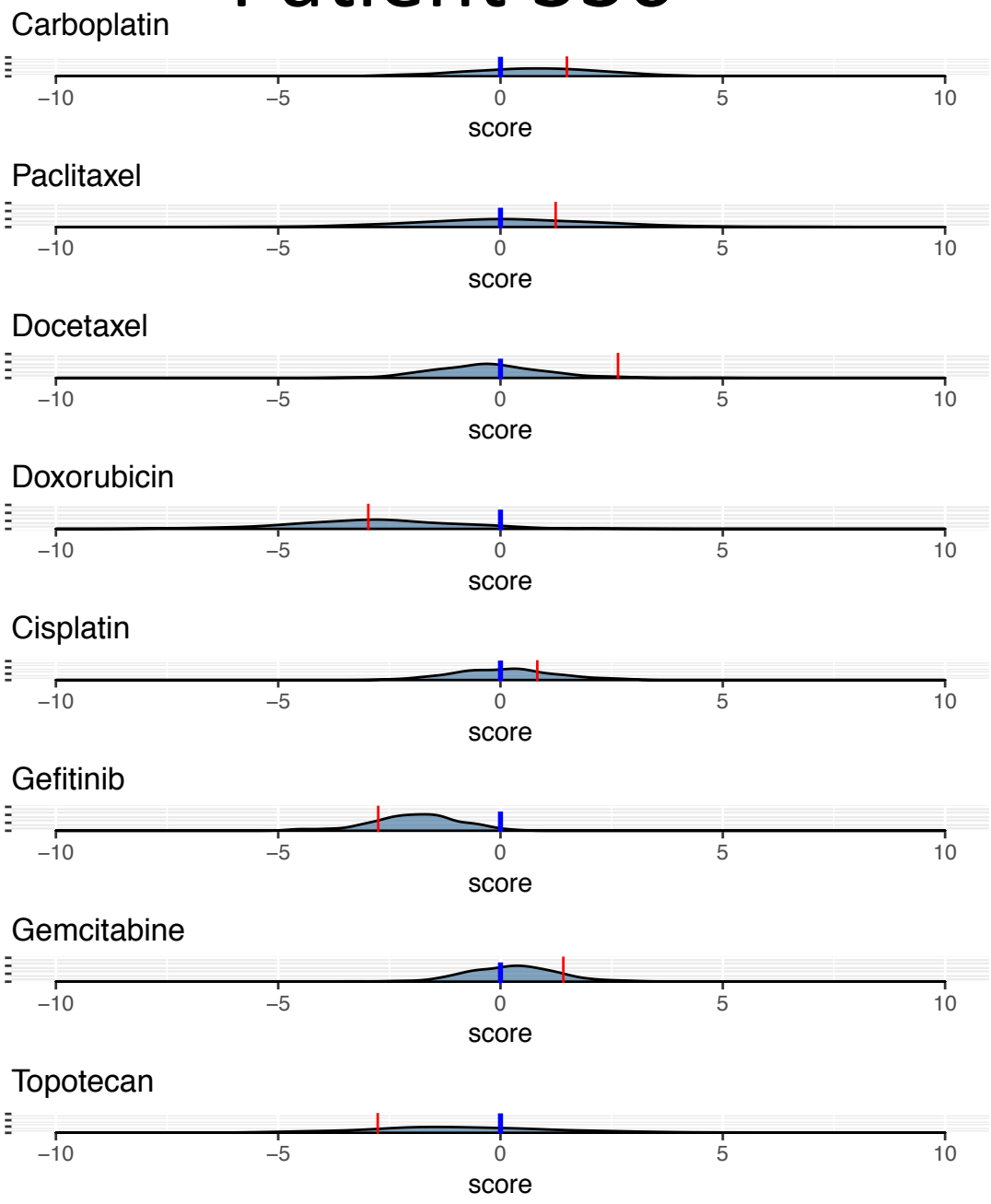
Patient 286


Carboplatin
−10
−5
0
5
10
score
Paclitaxel
−10
−5
0
5
10
score
Docetaxel
−10
−5
0
5
10
score
Doxorubicin
−10
−5
0
5
10
score
Cisplatin
−10
−5
0
5
10
score
Gefitinib
−10
−5
0
5
10
score
Gemcitabine
−10
−5
0
5
10
score
Topotecan
−10
−5
0
5
10
score

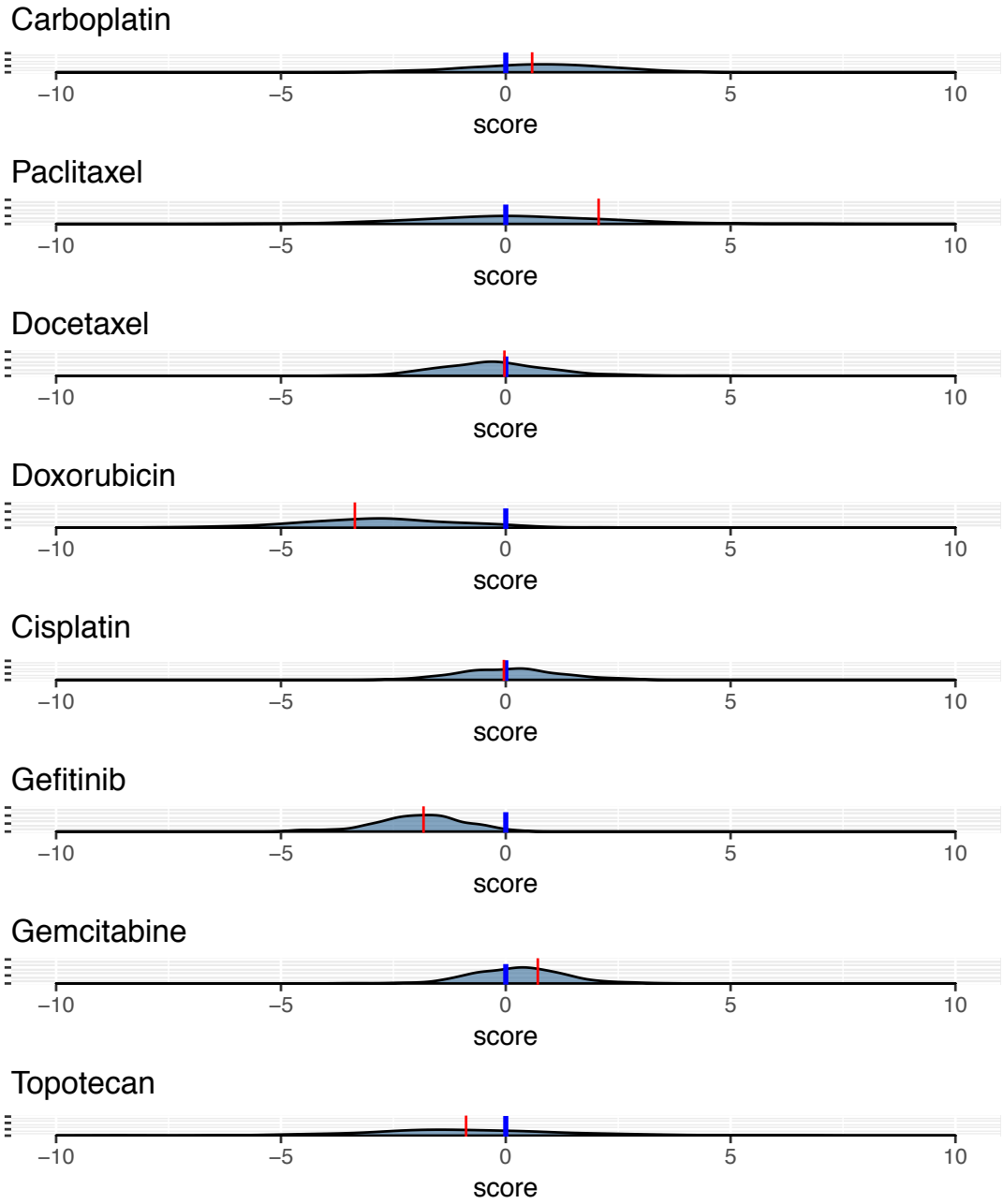
Patient 317


Carboplatin
−10
−5
0
5
10
score
Paclitaxel
−10
−5
0
5
10
score
Docetaxel
−10
−5
0
5
10
score
Doxorubicin
−10
−5
0
5
10
score
Cisplatin
−10
−5
0
5
10
score
Gefitinib
−10
−5
0
5
10
score
Gemcitabine
−10
−5
0
5
10
score
Topotecan
−10
−5
0
5
10
score

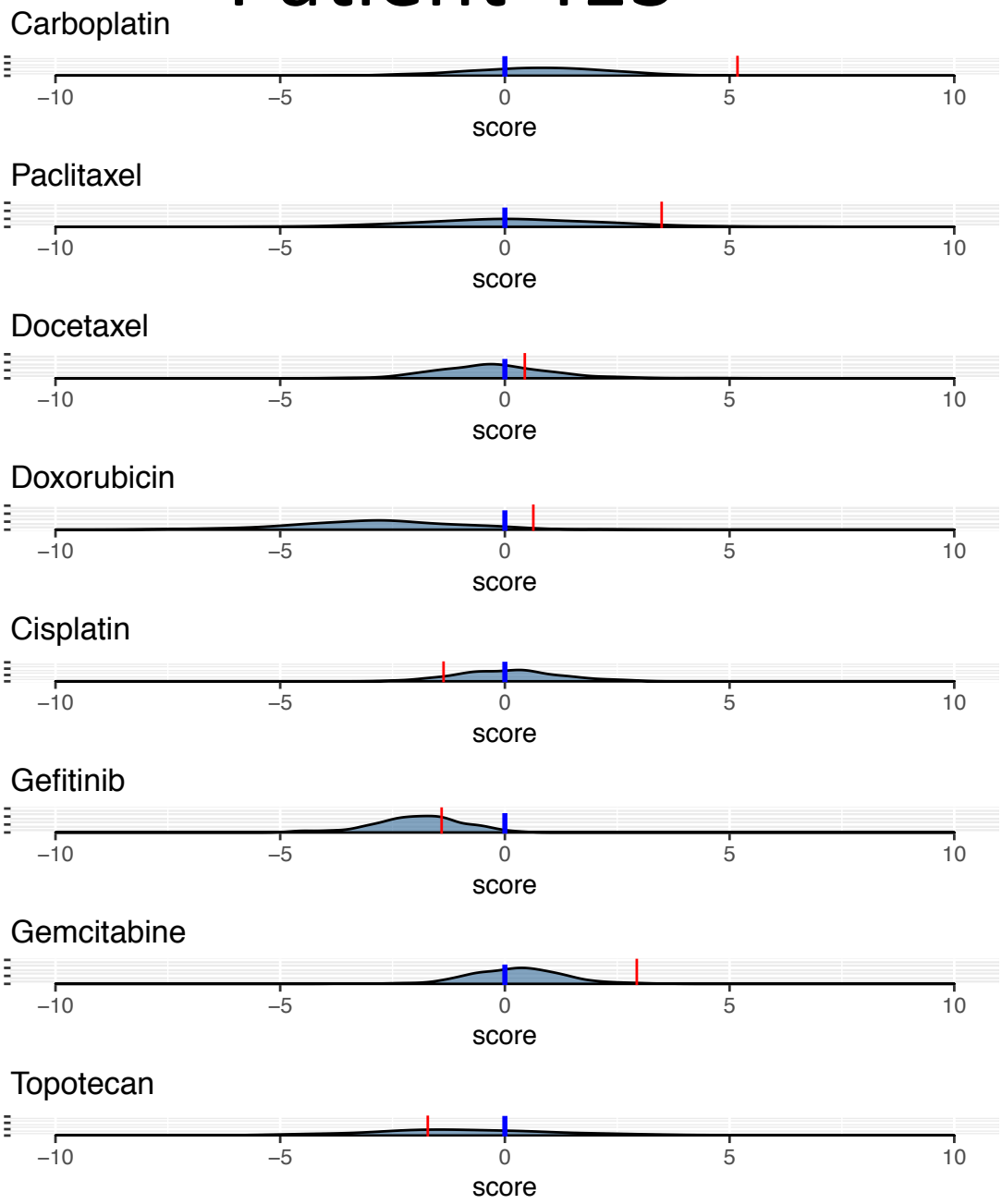
Patient 336


Carboplatin
−10
−5
0
5
10
score
Paclitaxel
−10
−5
0
5
10
score
Docetaxel
−10
−5
0
5
10
score
Doxorubicin
−10
−5
0
5
10
score
Cisplatin
−10
−5
0
5
10
score
Gefitinib
−10
−5
0
5
10
score
Gemcitabine
−10
−5
0
5
10
score
Topotecan
−10
−5
0
5
10
score

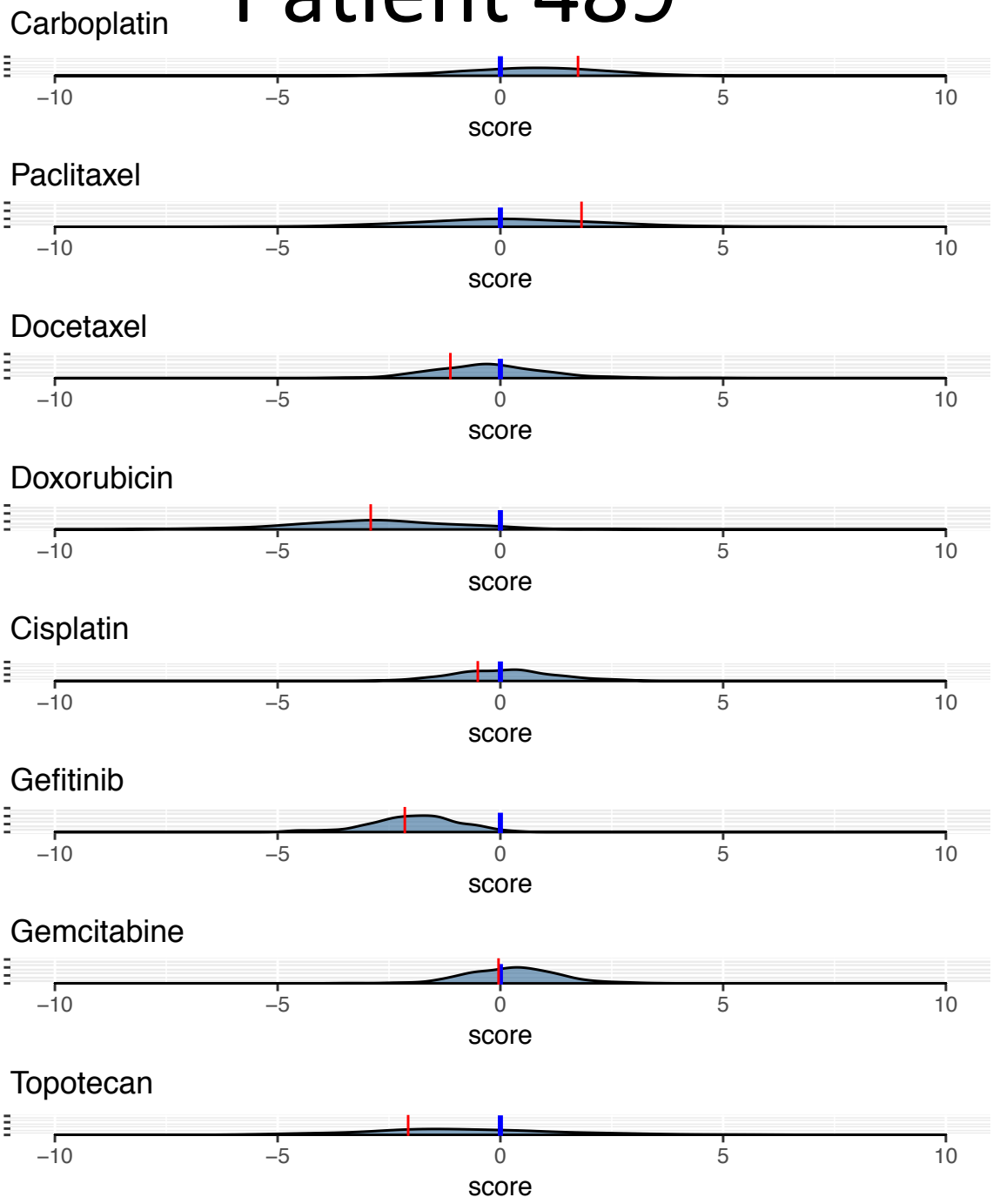
Patient 367


Carboplatin
−10
−5
0
5
10
score
Paclitaxel
−10
−5
0
5
10
score
Docetaxel
−10
−5
0
5
10
score
Doxorubicin
−10
−5
0
5
10
score
Cisplatin
−10
−5
0
5
10
score
Gefitinib
−10
−5
0
5
10
score
Gemcitabine
−10
−5
0
5
10
score
Topotecan
−10
−5
0
5
10
score

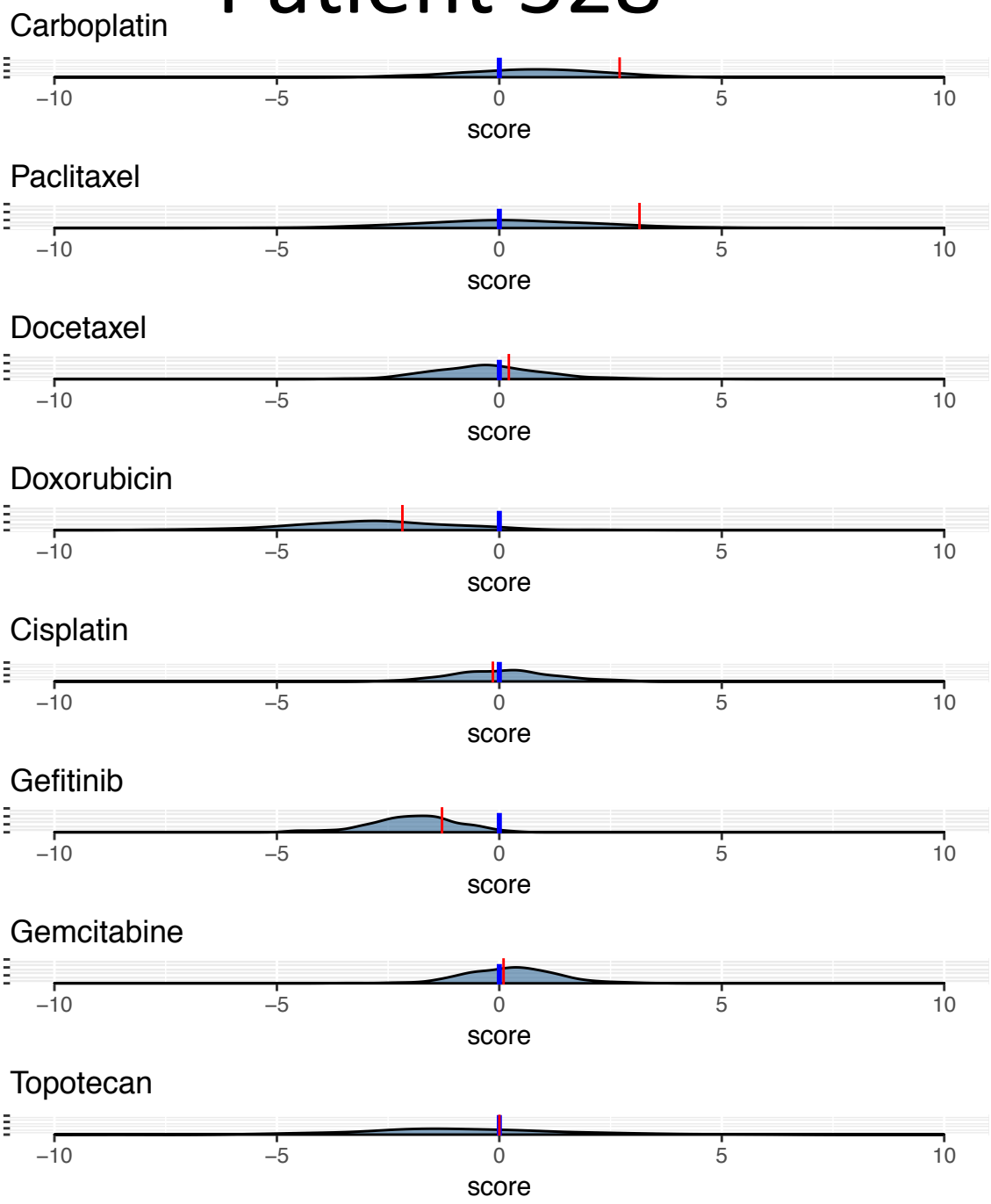
Patient 413


Carboplatin
−10
−5
0
5
10
score
Paclitaxel
−10
−5
0
5
10
score
Docetaxel
−10
−5
0
5
10
score
Doxorubicin
−10
−5
0
5
10
score
Cisplatin
−10
−5
0
5
10
score
Gefitinib
−10
−5
0
5
10
score
Gemcitabine
−10
−5
0
5
10
score
Topotecan
−10
−5
0
5
10
score

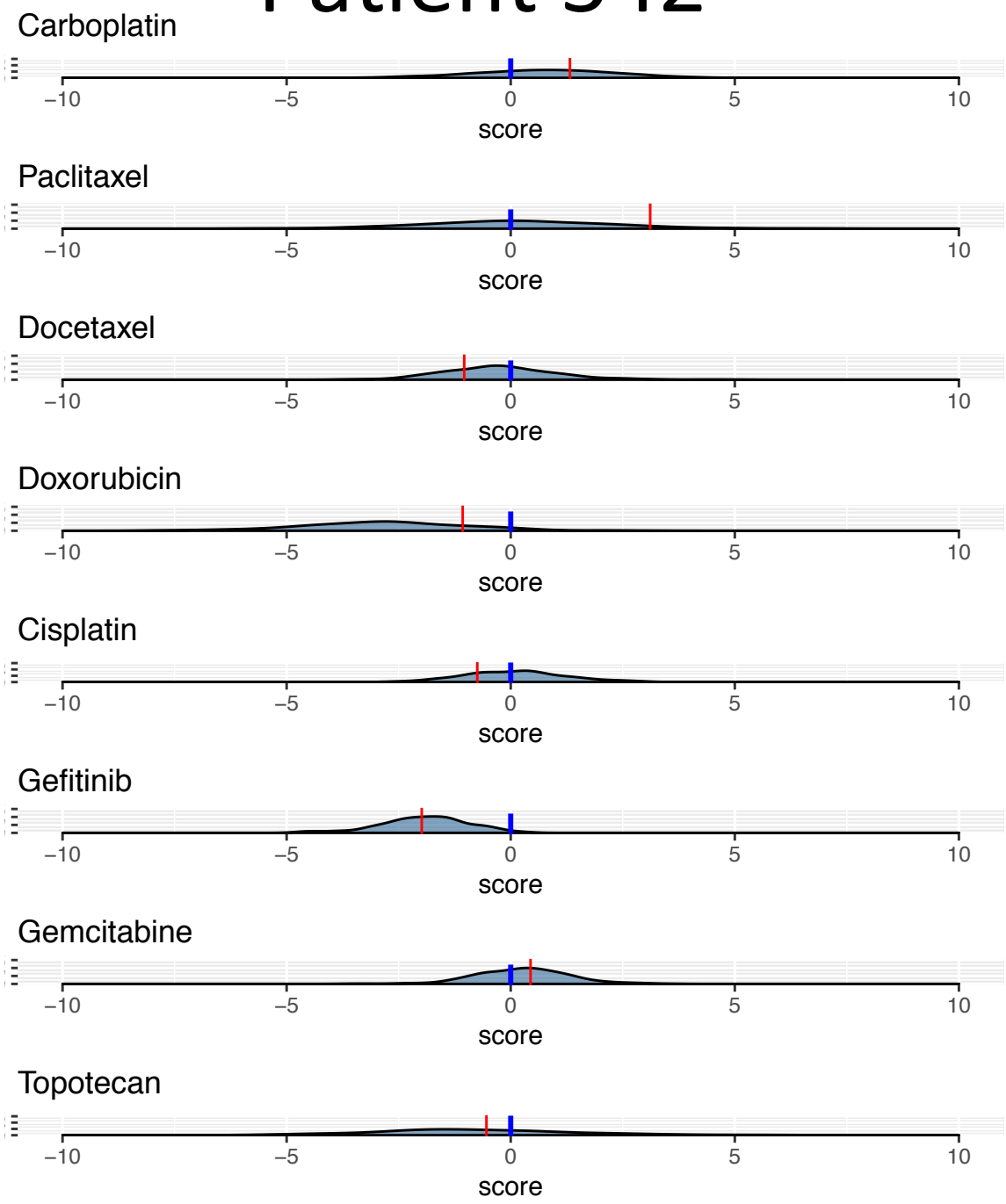
Patient 489


Carboplatin
−10
−5
0
5
10
score
Paclitaxel
−10
−5
0
5
10
score
Docetaxel
−10
−5
0
5
10
score
Doxorubicin
−10
−5
0
5
10
score
Cisplatin
−10
−5
0
5
10
score
Gefitinib
−10
−5
0
5
10
score
Gemcitabine
−10
−5
0
5
10
score
Topotecan
−10
−5
0
5
10
score

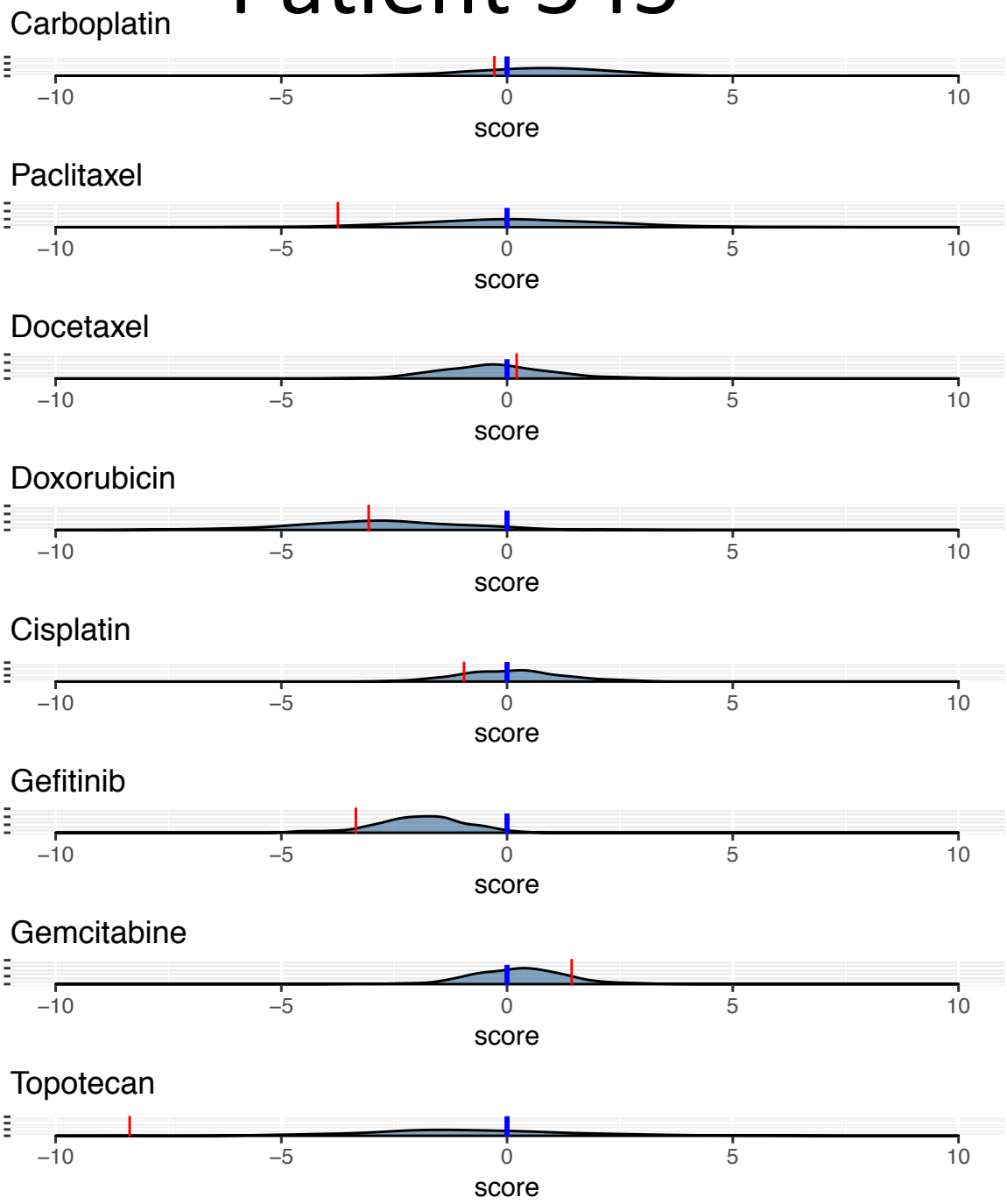
Patient 528


Carboplatin
−10
−5
0
5
10
score
Paclitaxel
−10
−5
0
5
10
score
Docetaxel
−10
−5
0
5
10
score
Doxorubicin
−10
−5
0
5
10
score
Cisplatin
−10
−5
0
5
10
score
Gefitinib
−10
−5
0
5
10
score
Gemcitabine
−10
−5
0
5
10
score
Topotecan
−10
−5
0
5
10
score

Patient 542

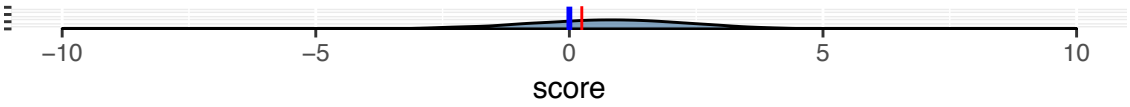

Carboplatin
−10
−5
0
5
10
score
Paclitaxel
−10
−5
0
5
10
score
Docetaxel
−10
−5
0
5
10
score
Doxorubicin
−10
−5
0
5
10
score
Cisplatin
−10
−5
0
5
10
score
Gefitinib
−10
−5
0
5
10
score
Gemcitabine
−10
−5
0
5
10
score
Topotecan
−10
−5
0
5
10
score

Patient 545

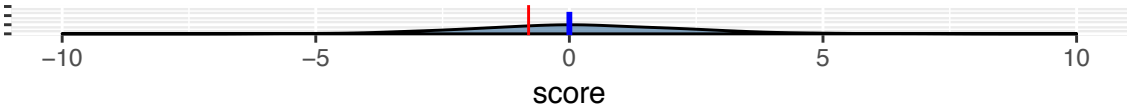

Carboplatin
−10
−5
0
5
10
score
Paclitaxel
−10
−5
0
5
10
score
Docetaxel
−10
−5
0
5
10
score
Doxorubicin
−10
−5
0
5
10
score
Cisplatin
−10
−5
0
5
10
score
Gefitinib
−10
−5
0
5
10
score
Gemcitabine
−10
−5
0
5
10
score
Topotecan
−10
−5
0
5
10
score

Patient 588

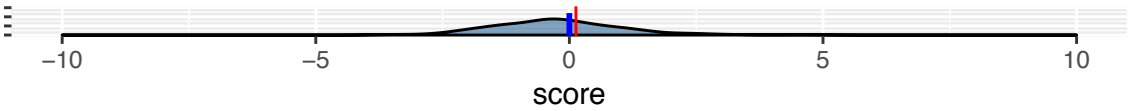
Carboplatin



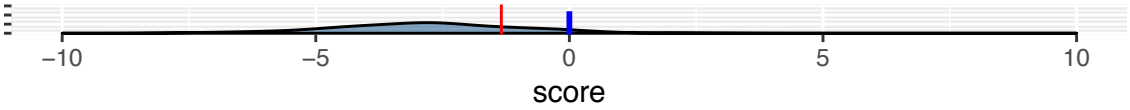
Paclitaxel



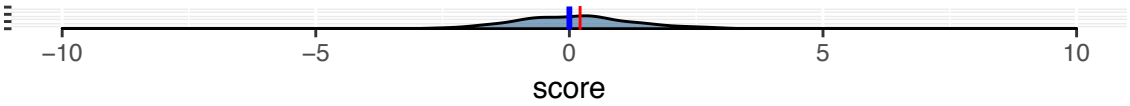
Docetaxel



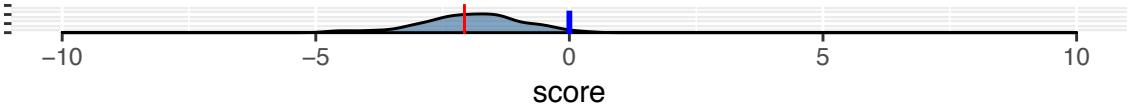
Doxorubicin



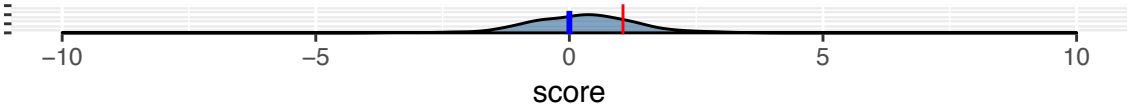
Cisplatin



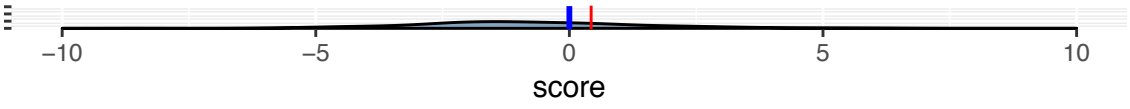
Gefitinib



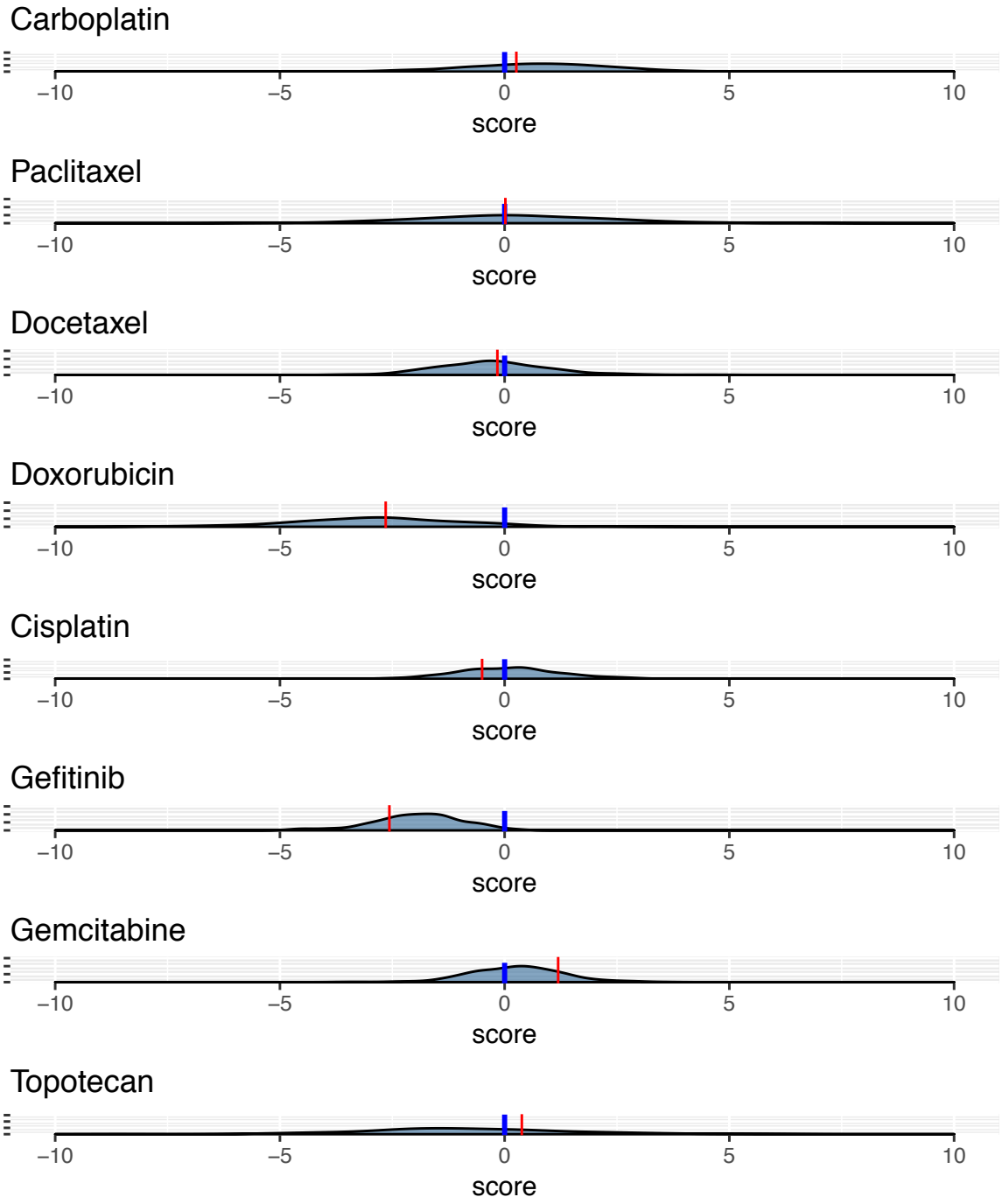
Gemcitabine



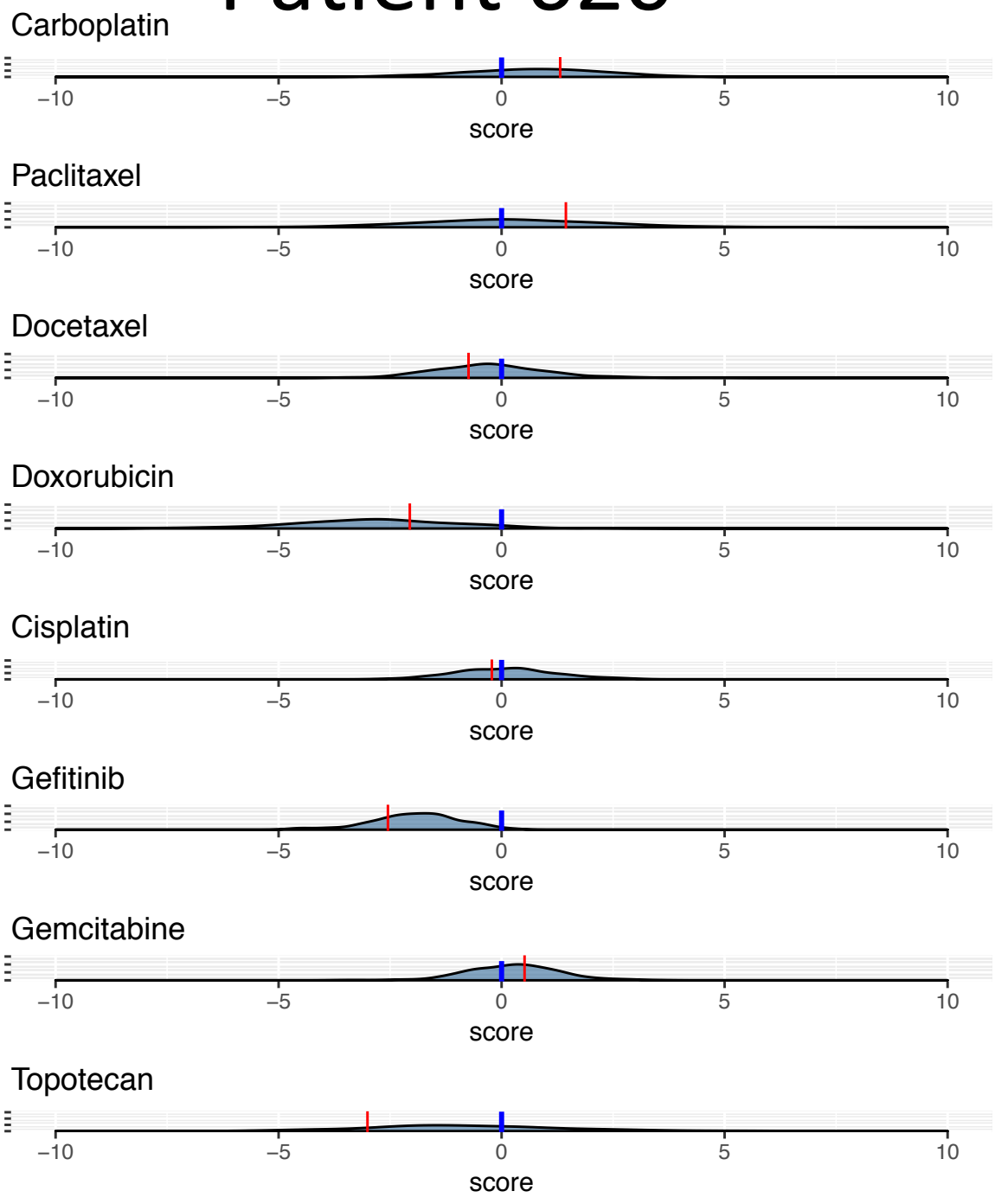
Topotecan



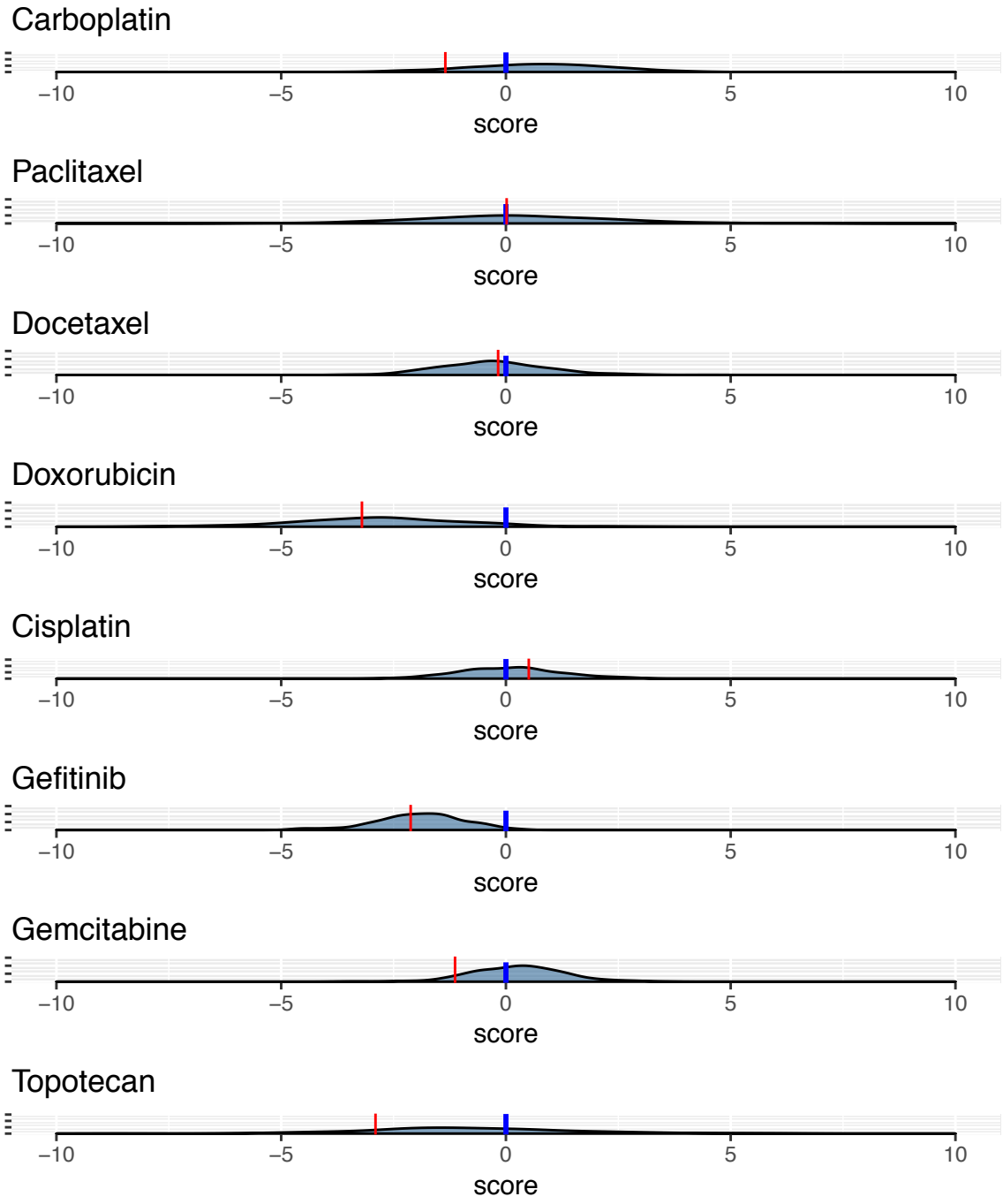
Patient 617


Carboplatin
−10
−5
0
5
10
score
Paclitaxel
−10
−5
0
5
10
score
Docetaxel
−10
−5
0
5
10
score
Doxorubicin
−10
−5
0
5
10
score
Cisplatin
−10
−5
0
5
10
score
Gefitinib
−10
−5
0
5
10
score
Gemcitabine
−10
−5
0
5
10
score
Topotecan
−10
−5
0
5
10
score

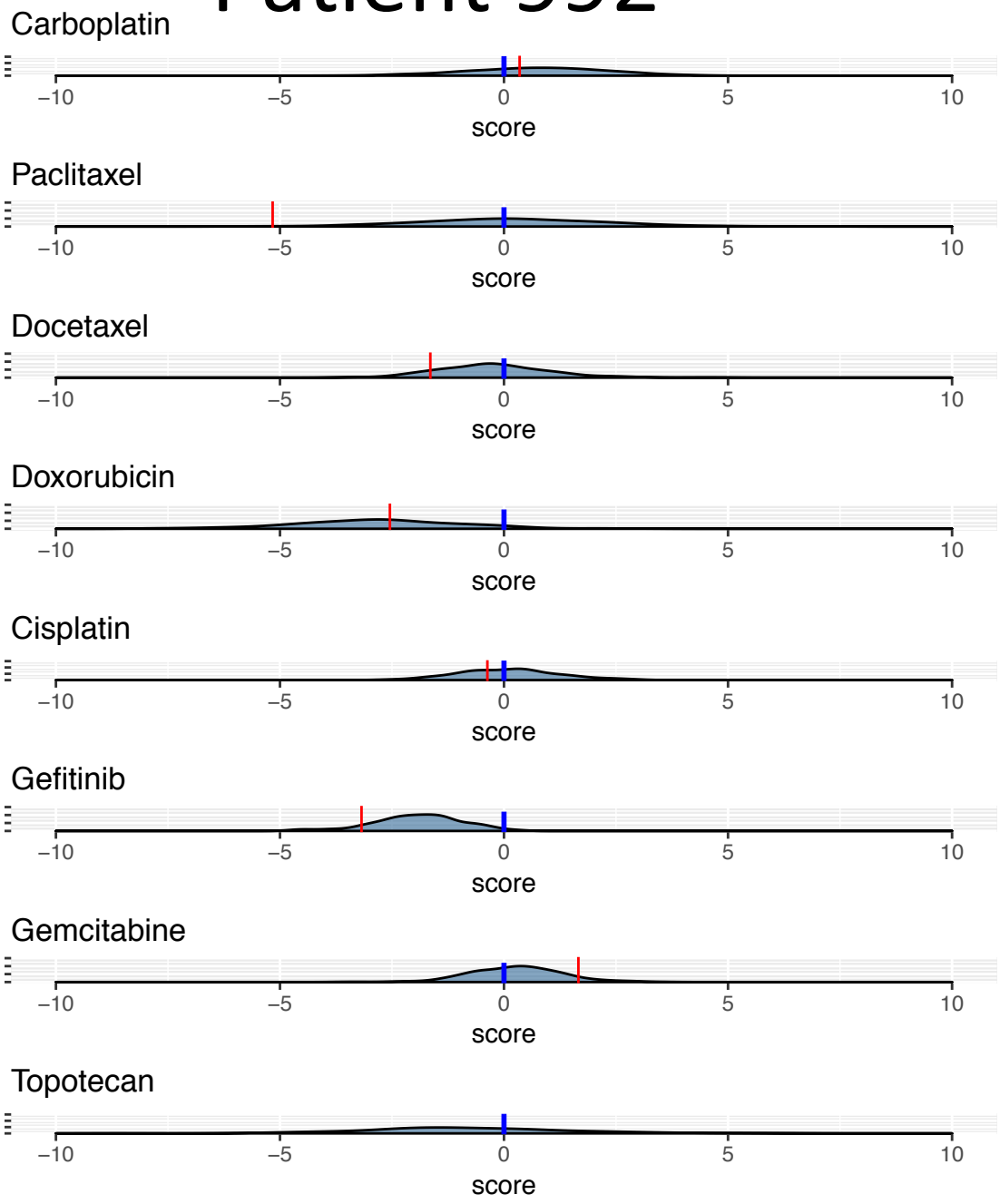
Patient 620


Carboplatin
−10
−5
0
5
10
score
Paclitaxel
−10
−5
0
5
10
score
Docetaxel
−10
−5
0
5
10
score
Doxorubicin
−10
−5
0
5
10
score
Cisplatin
−10
−5
0
5
10
score
Gefitinib
−10
−5
0
5
10
score
Gemcitabine
−10
−5
0
5
10
score
Topotecan
−10
−5
0
5
10
score

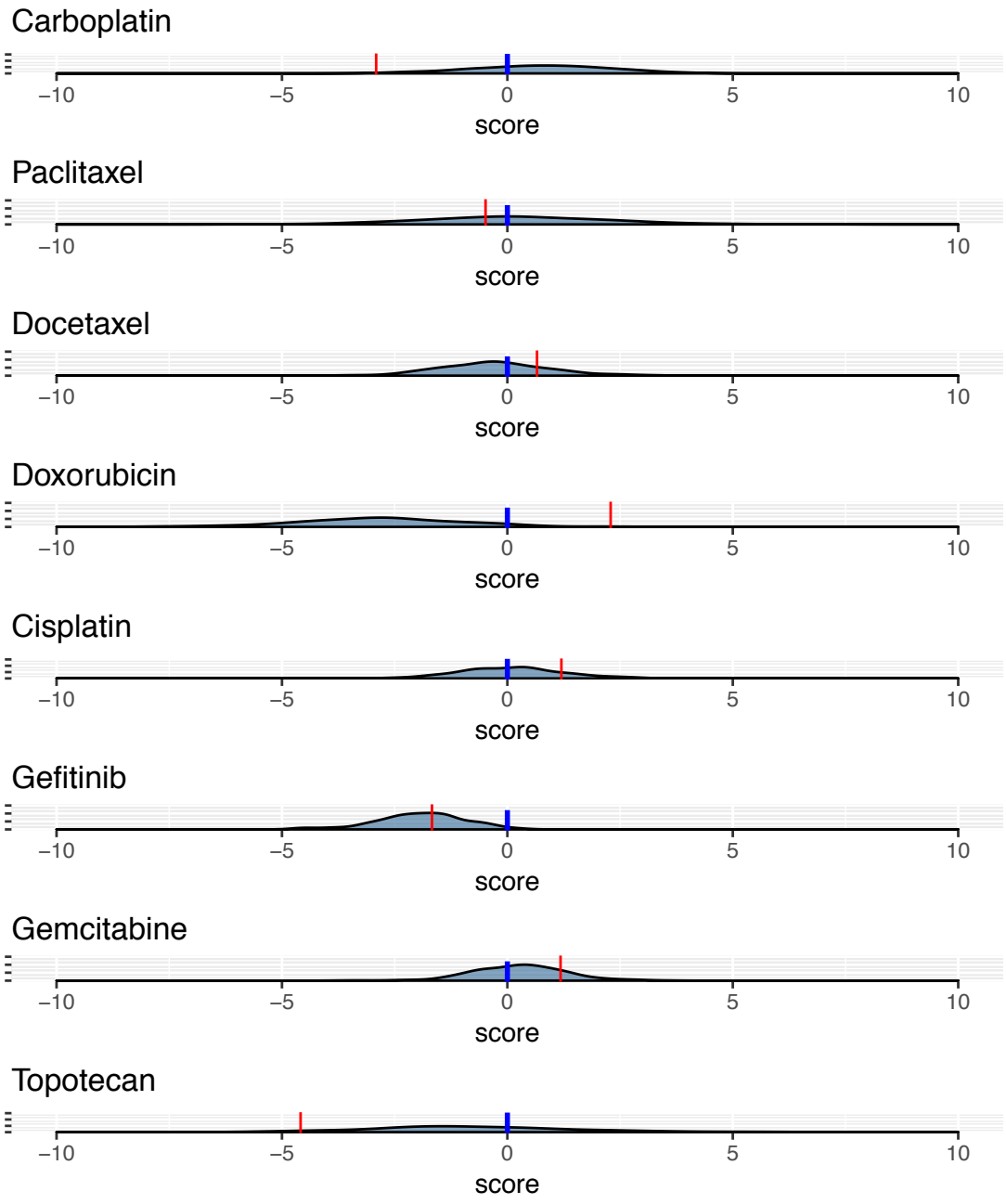
Patient 813


Carboplatin
−10
−5
0
5
10
score
Paclitaxel
−10
−5
0
5
10
score
Docetaxel
−10
−5
0
5
10
score
Doxorubicin
−10
−5
0
5
10
score
Cisplatin
−10
−5
0
5
10
score
Gefitinib
−10
−5
0
5
10
score
Gemcitabine
−10
−5
0
5
10
score
Topotecan
−10
−5
0
5
10
score

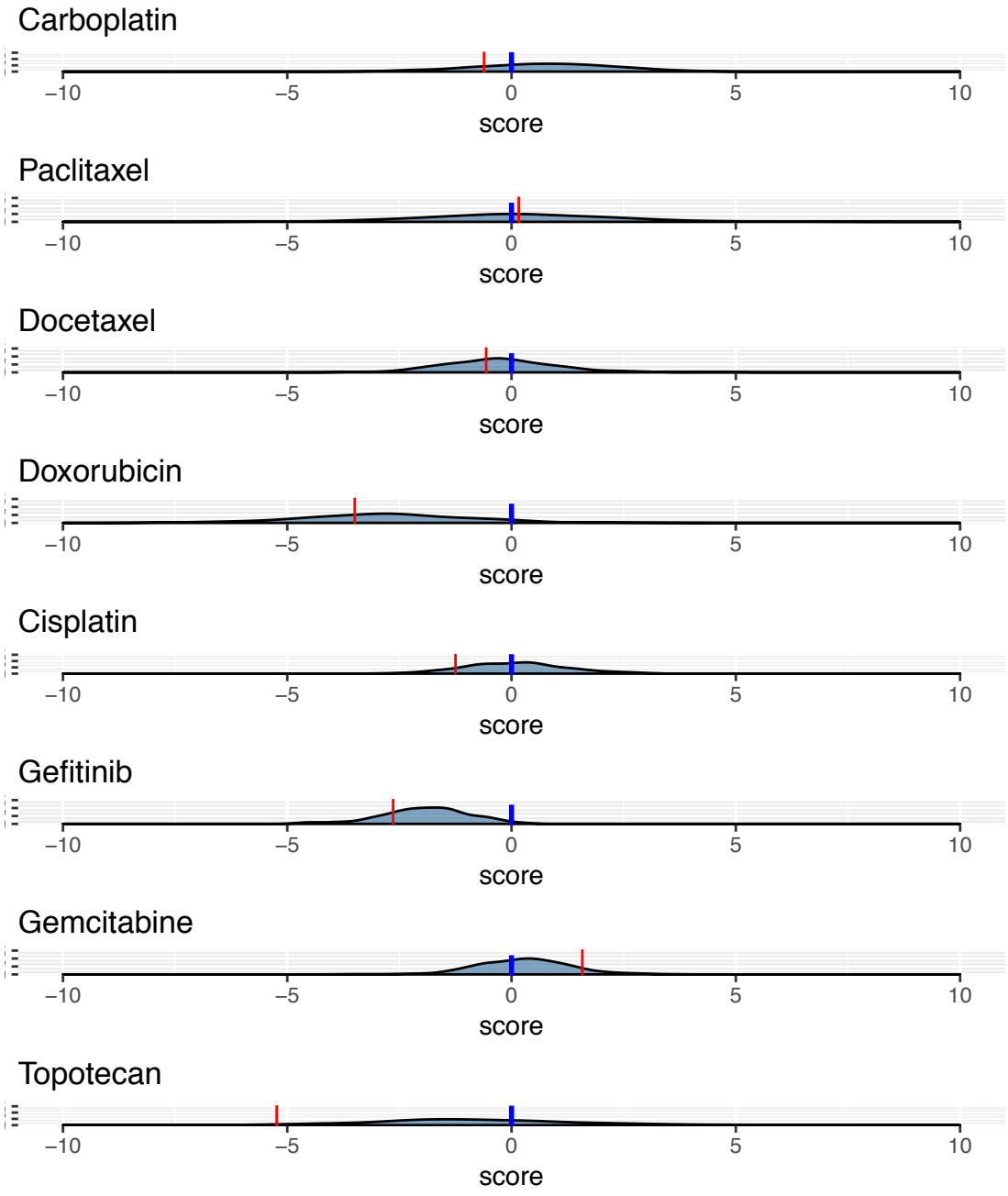
Patient 992


Carboplatin
−10
−5
0
5
10
score
Paclitaxel
−10
−5
0
5
10
score
Docetaxel
−10
−5
0
5
10
score
Doxorubicin
−10
−5
0
5
10
score
Cisplatin
−10
−5
0
5
10
score
Gefitinib
−10
−5
0
5
10
score
Gemcitabine
−10
−5
0
5
10
score
Topotecan
−10
−5
0
5
10
score

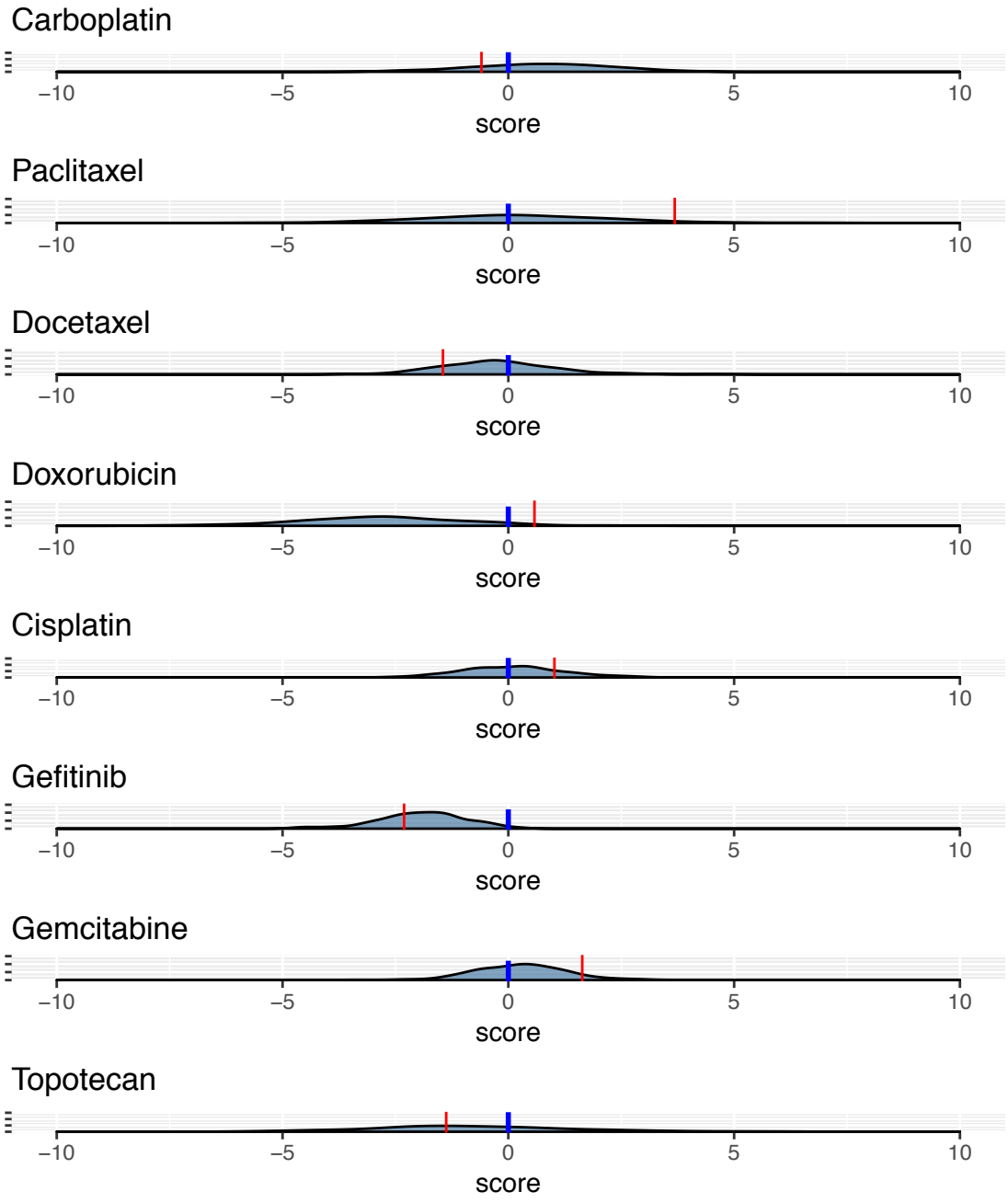
Patient 1012



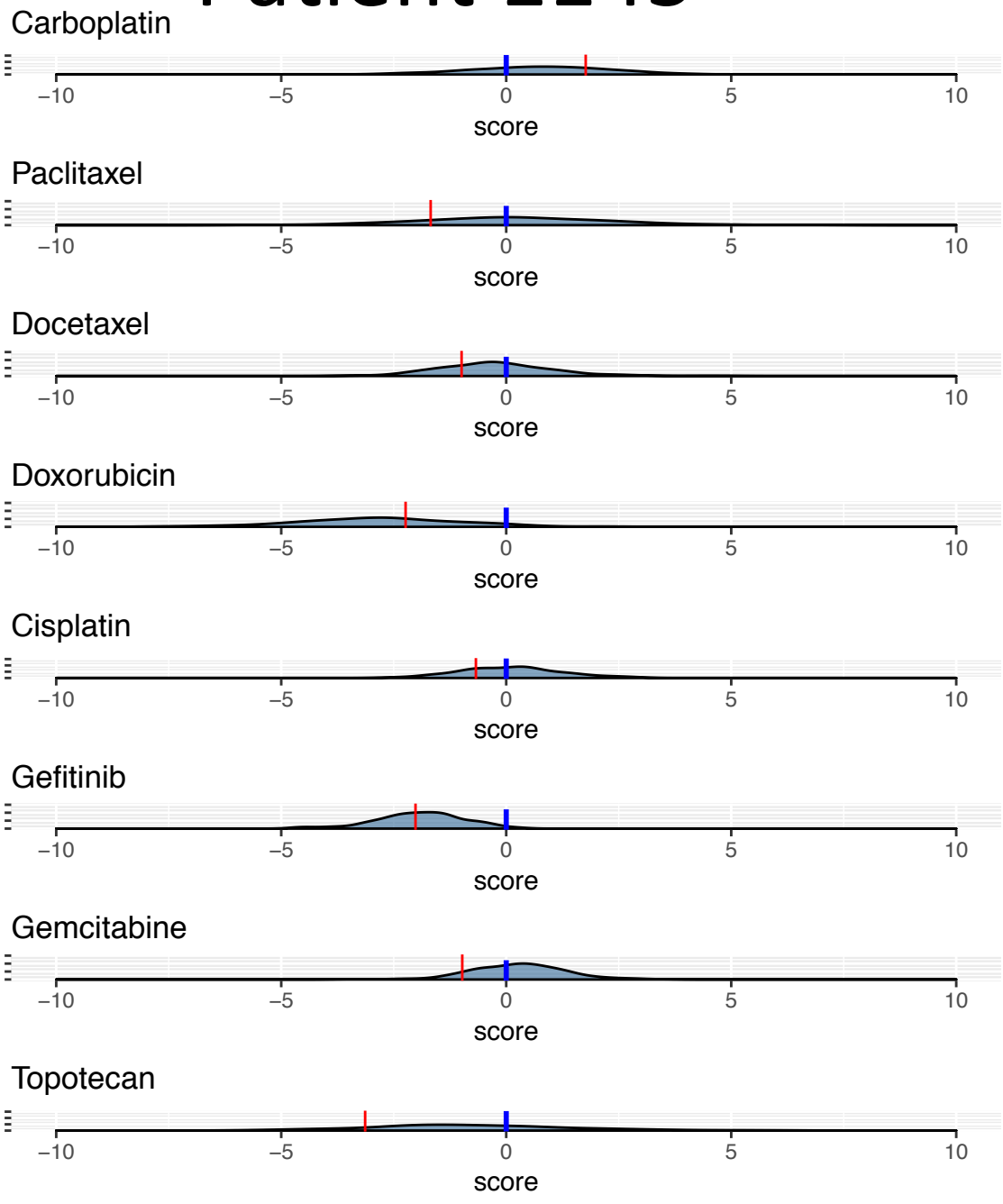
Patient 1122



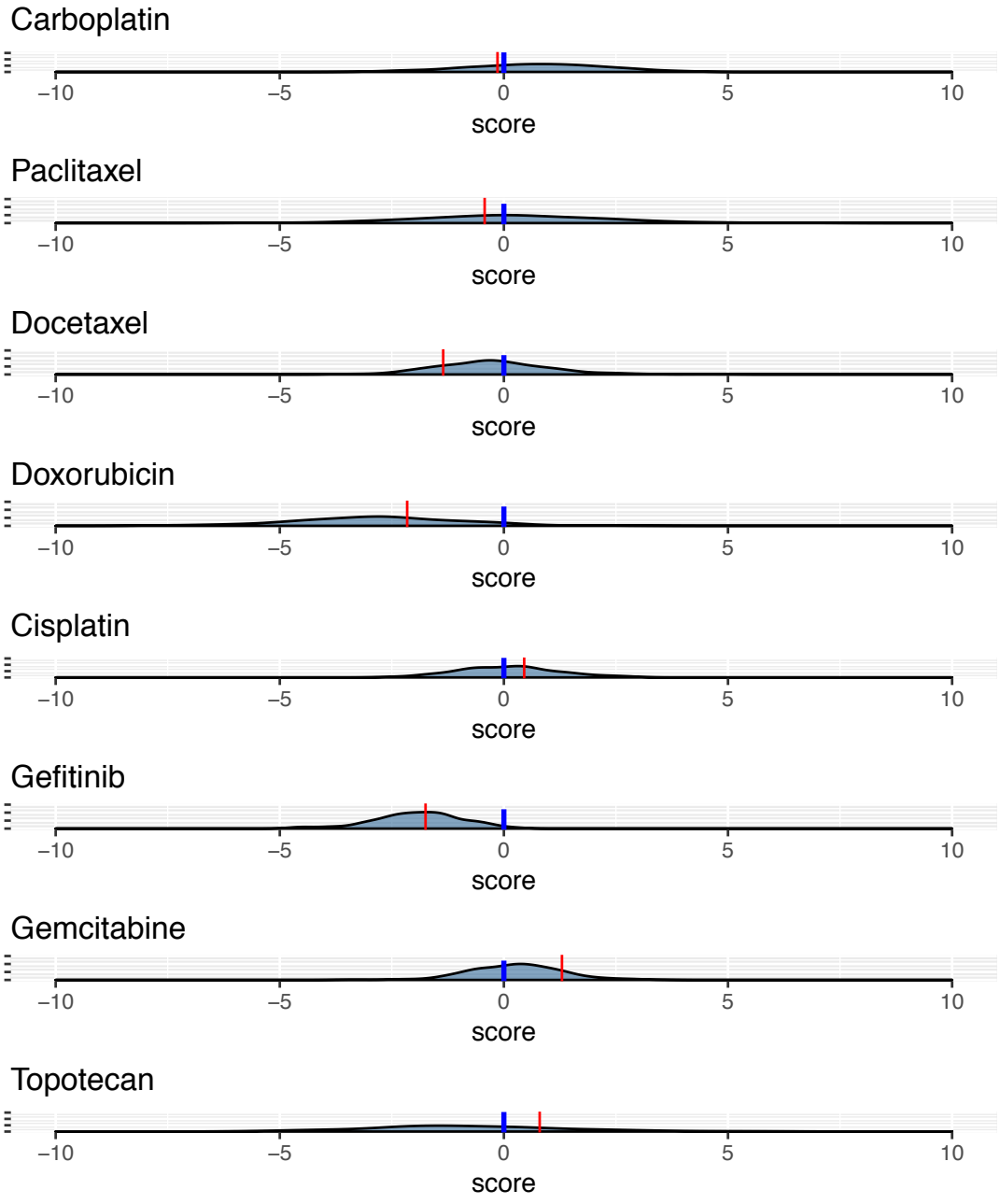
Patient 1129



Patient 1145

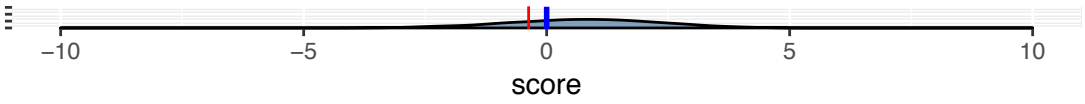


Patient BJ1

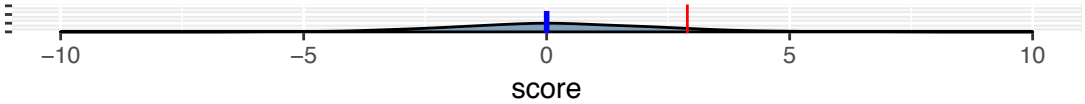

Carboplatin
−10
−5
0
5
10
score
Paclitaxel
−10
−5
0
5
10
score
Docetaxel
−10
−5
0
5
10
score
Doxorubicin
−10
−5
0
5
10
score
Cisplatin
−10
−5
0
5
10
score
Gefitinib
−10
−5
0
5
10
score
Gemcitabine
−10
−5
0
5
10
score
Topotecan
−10
−5
0
5
10
score

Patient BJ4

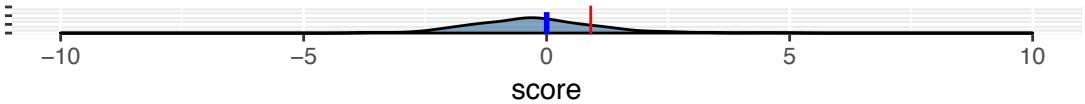
Carboplatin



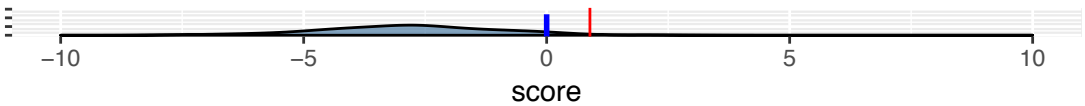
Paclitaxel



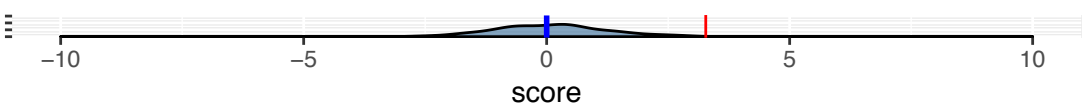
Docetaxel



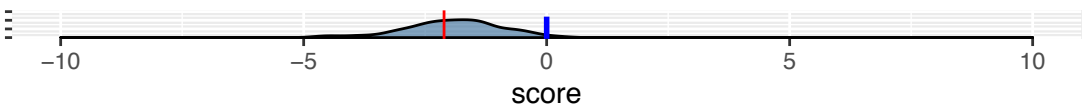
Doxorubicin



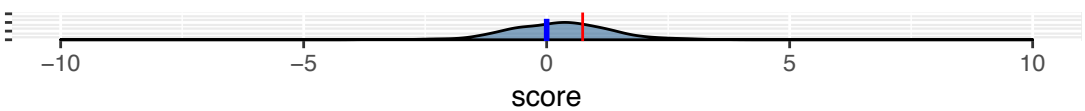
Cisplatin



Gefitinib



Gemcitabine



Topotecan

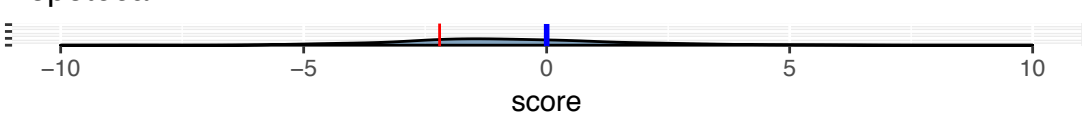
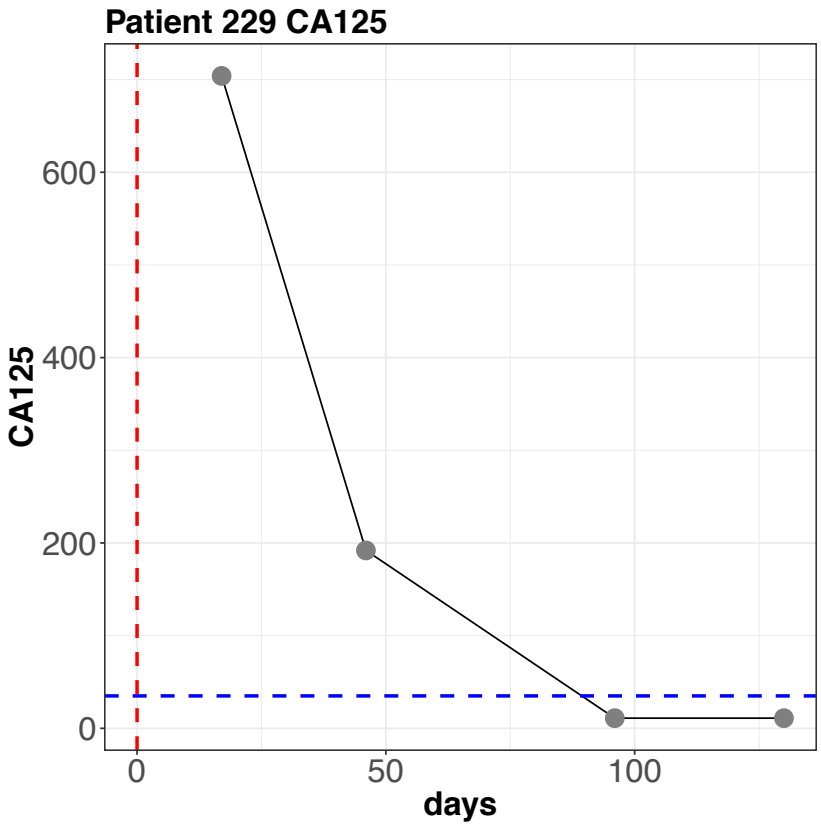
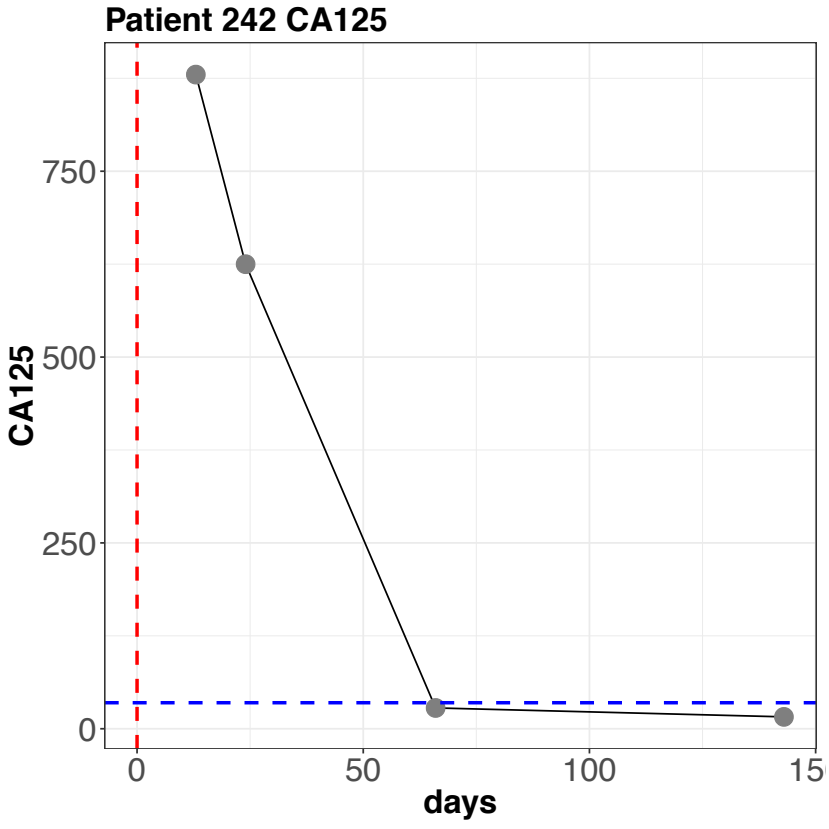
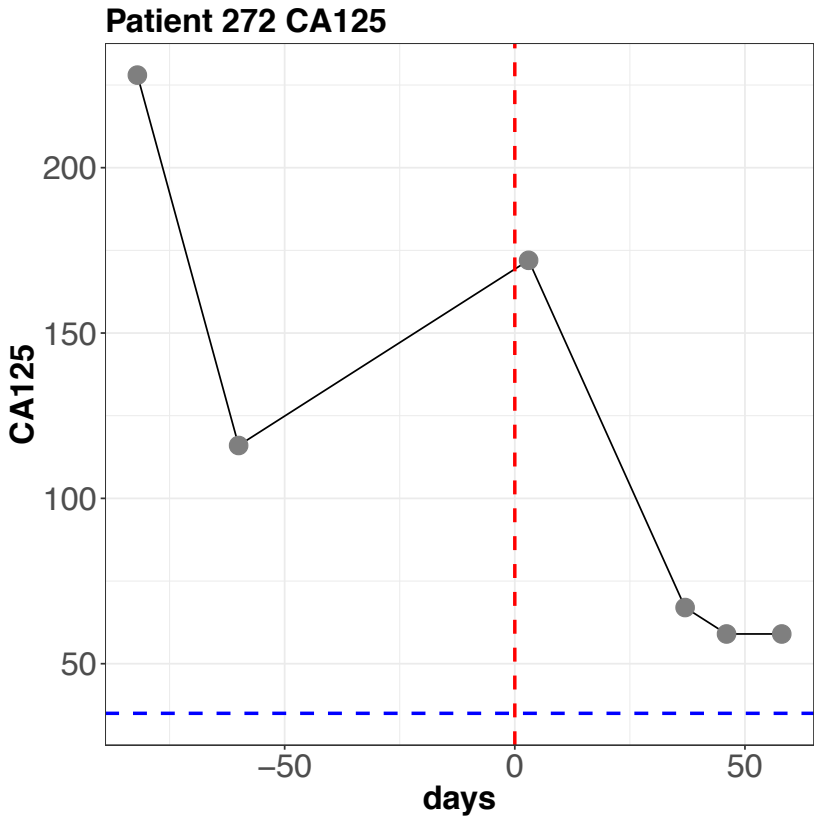
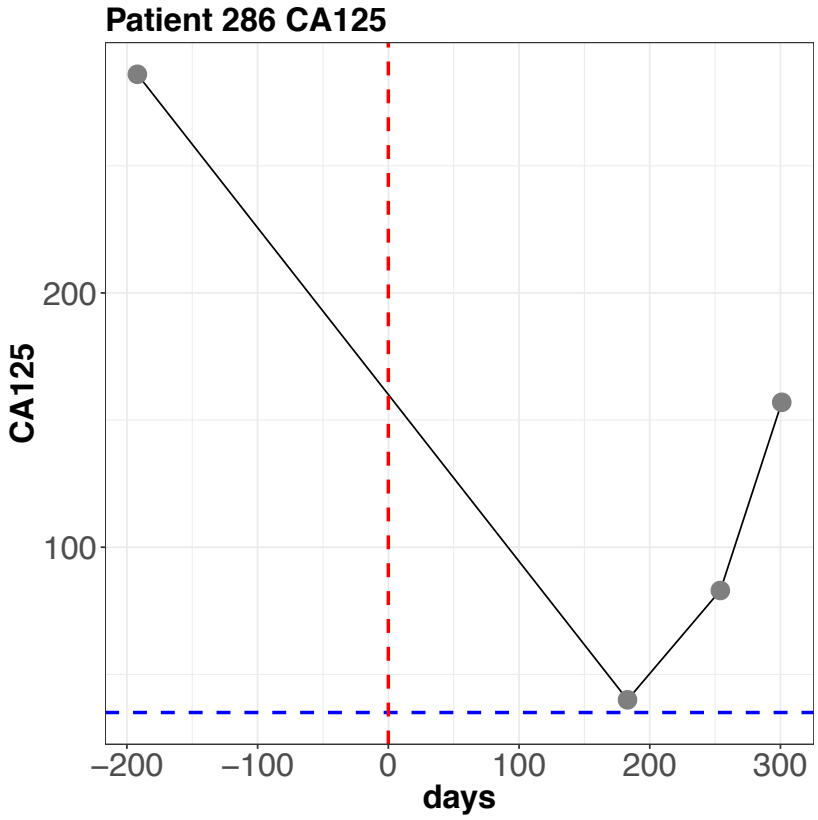


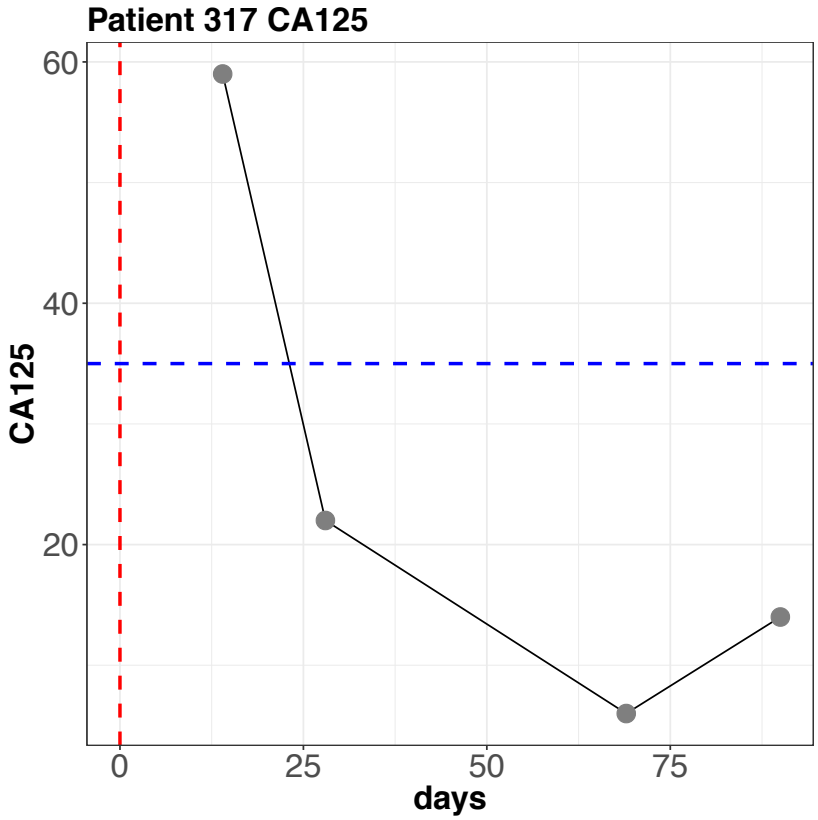
Figure 35 – The predicted response scores of each of the 23 ovarian cancer patients analyzed in this study (red lines) are plotted over the distribution of previously the predicted scores of 273 ovarian cancer patients [6].

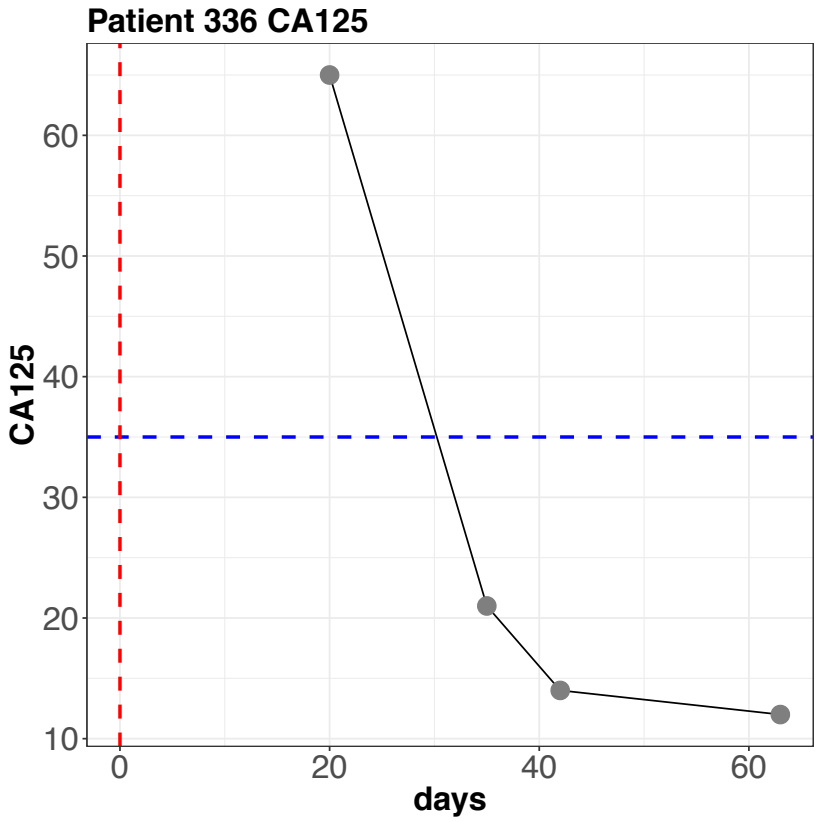

Patient 229 CA125
CA125
600
400
200
0
0
50
100
days

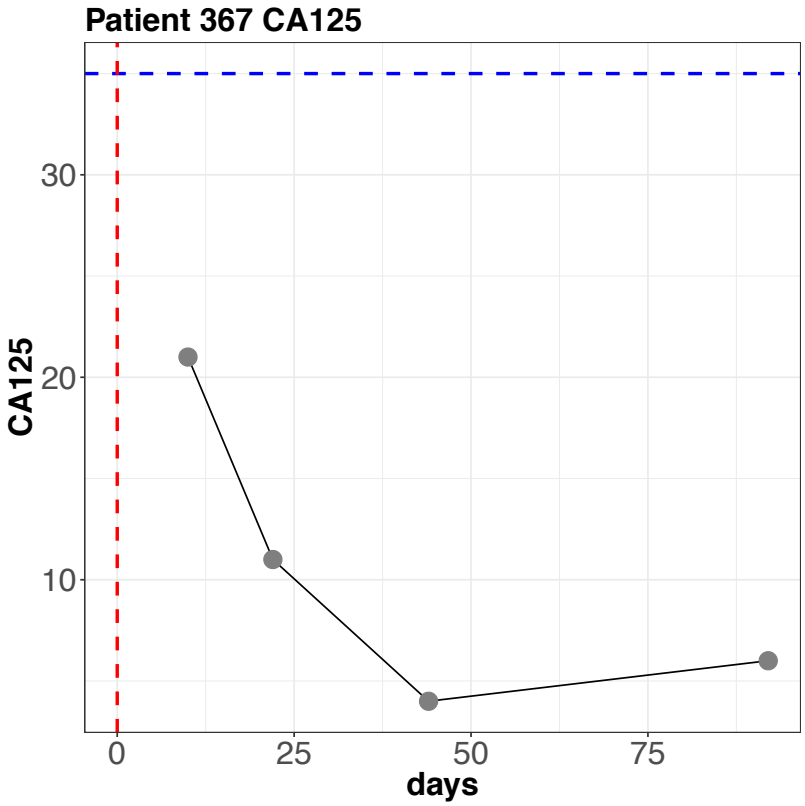

Patient 242 CA125
CA125
750
500
250
0
0
50
100
15
days

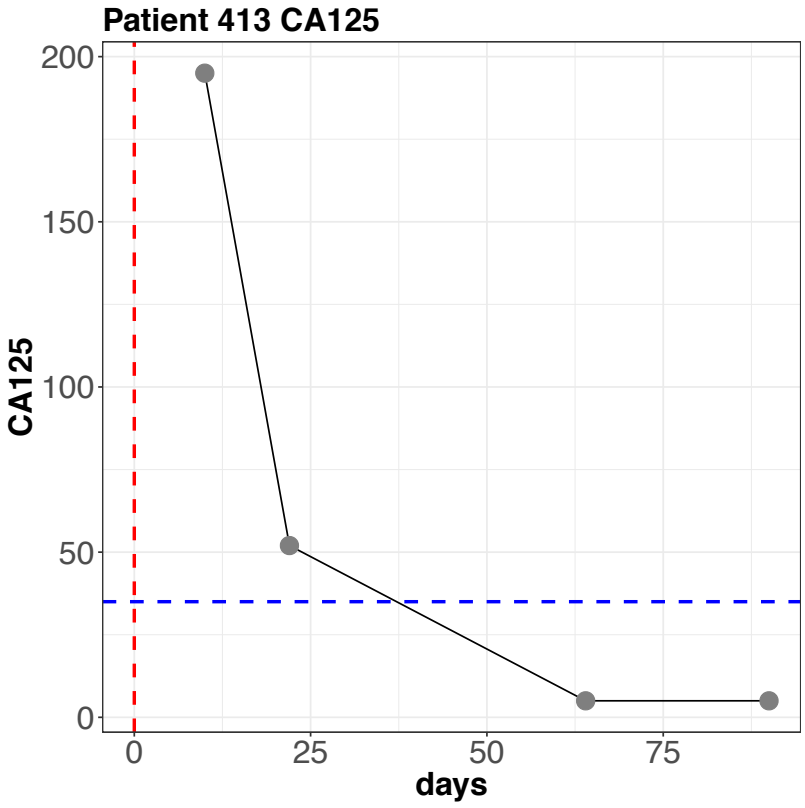

Patient 272 CA125
CA125
200
150
100
50
−50
0
50
days

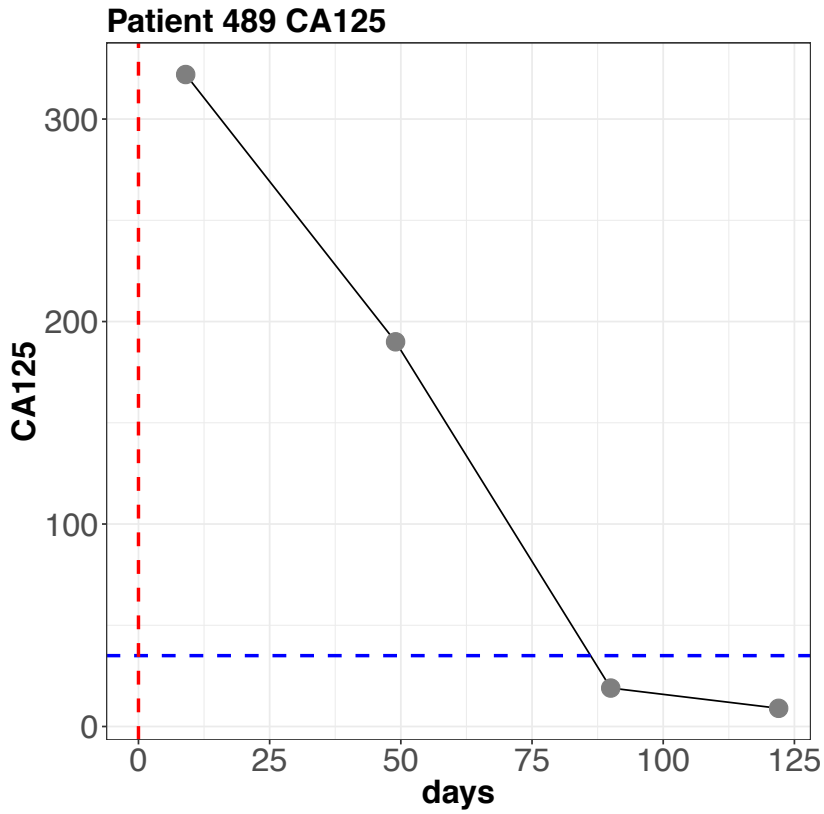

Patient 286 CA125
CA125
200
100
−200
−100
0
100
200
300
days

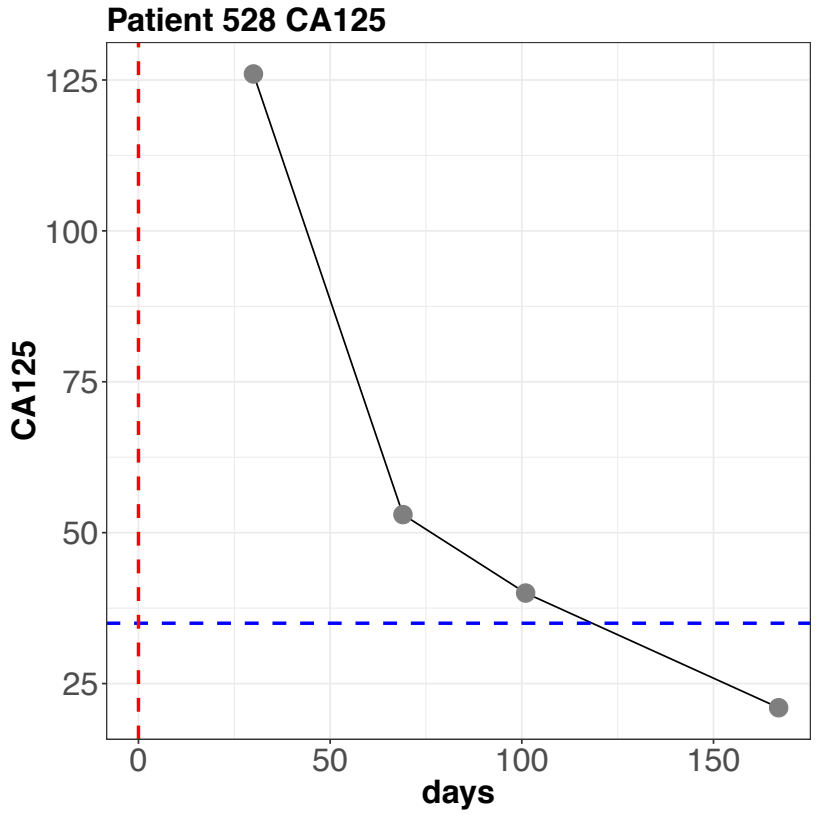

Patient 317 CA125
60
40
20
CA125
0
25
50
75
days

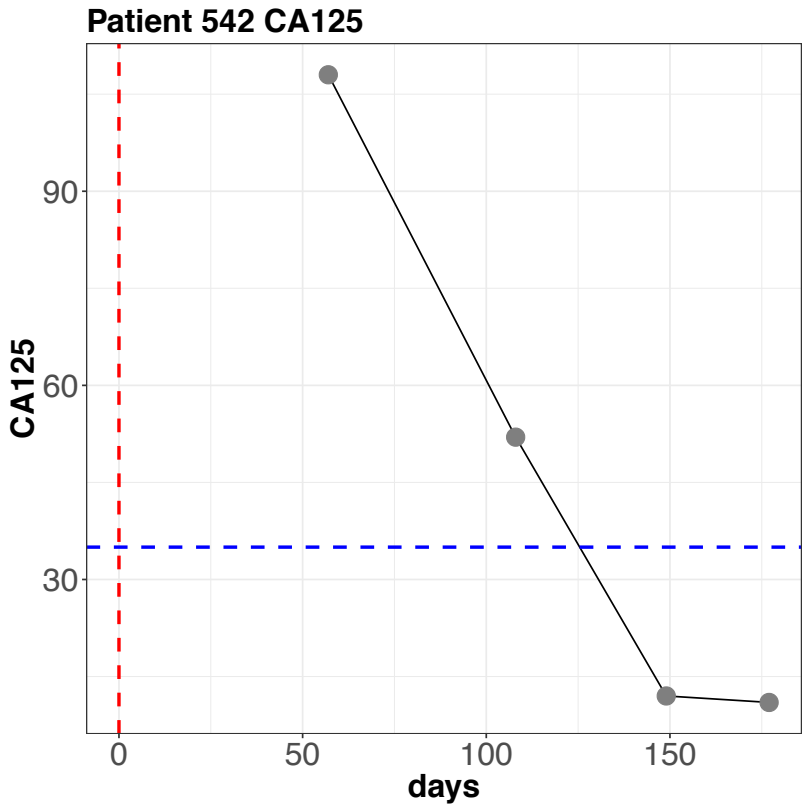

Patient 336 CA125
60
50
40
30
20
10
CA125
0
20
40
60
days

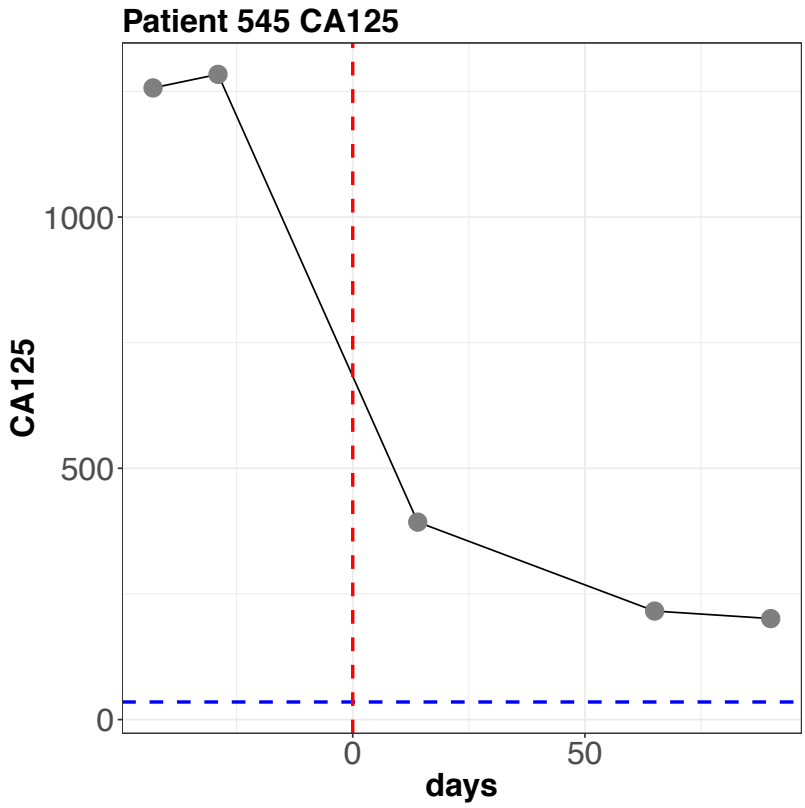

Patient 367 CA125
CA125
30
20
10
0
25
50
75
days

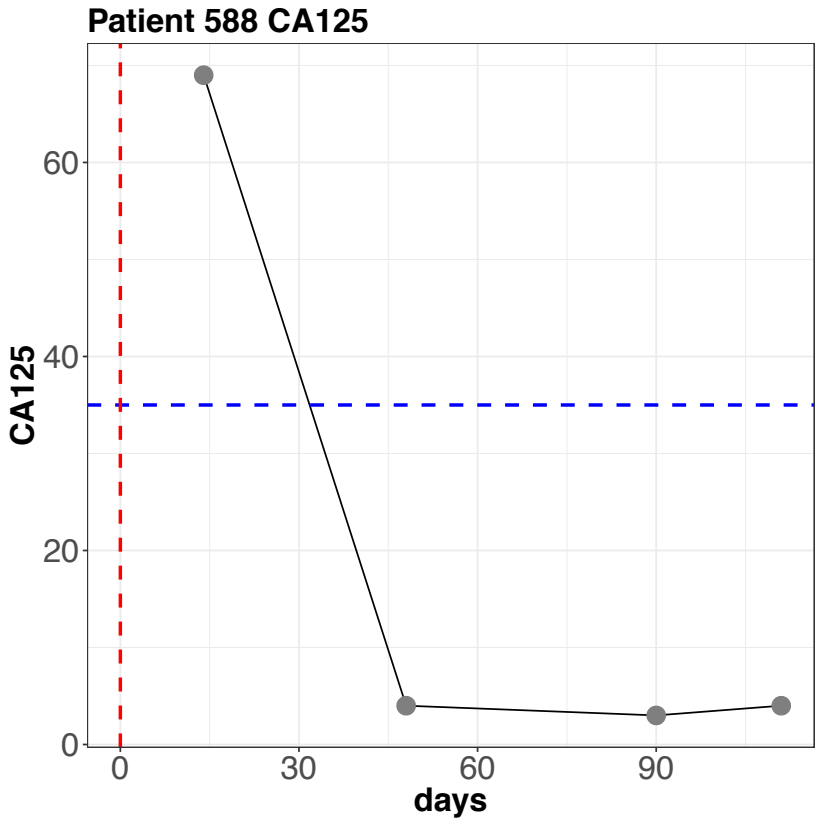

Patient 413 CA125
CA125
200
150
100
50
0
0
25
50
75
days

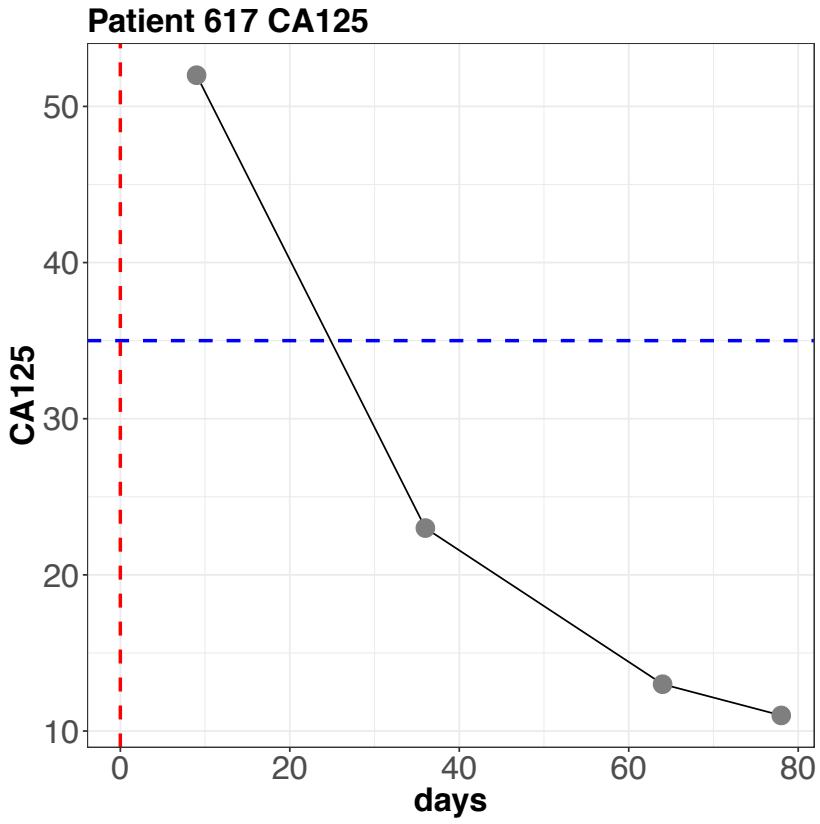

Patient 489 CA125
CA125
300
200
100
0
0
25
50
75
100
125
days

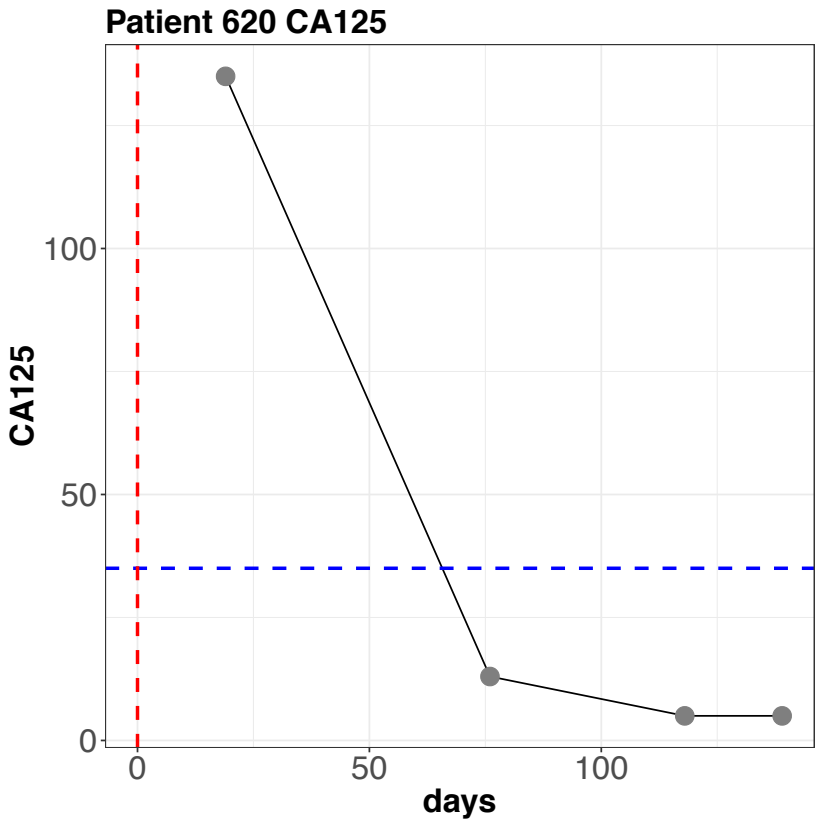

Patient 528 CA125
CA125
125
100
75
50
25
0
50
100
150
days

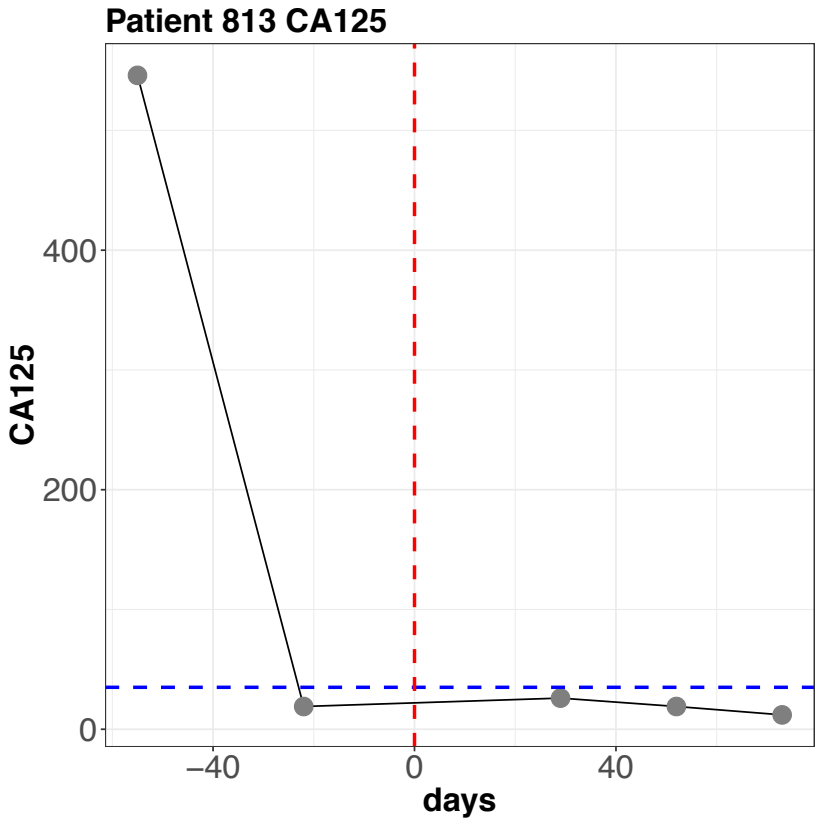

Patient 542 CA125
CA125
90
60
30
0
50
100
150
days

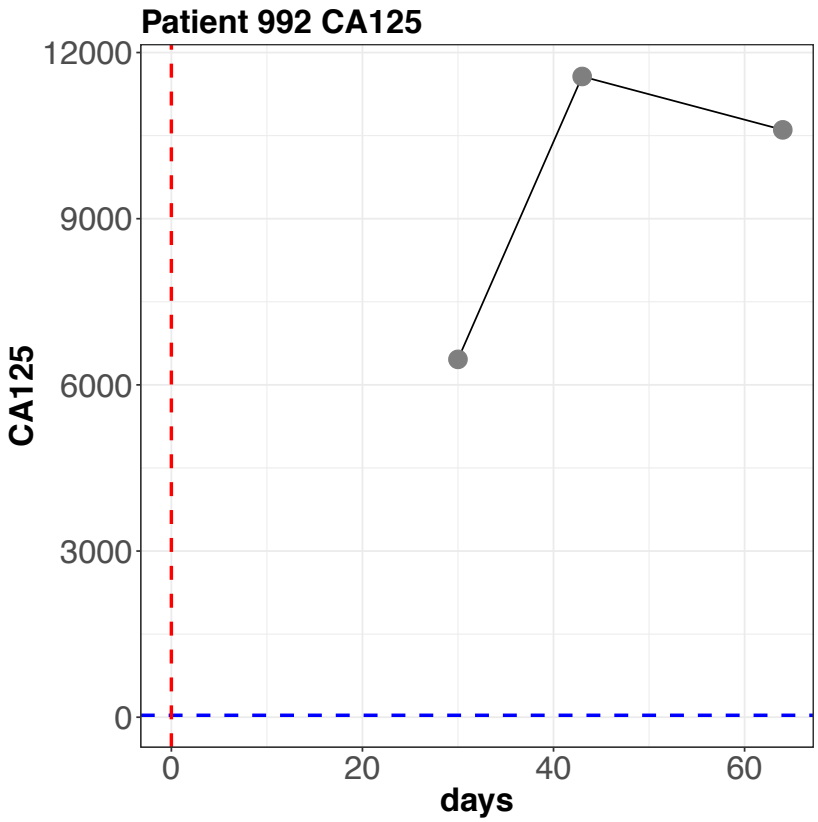

Patient 545 CA125
CA125
1000
500
0
0
50
days

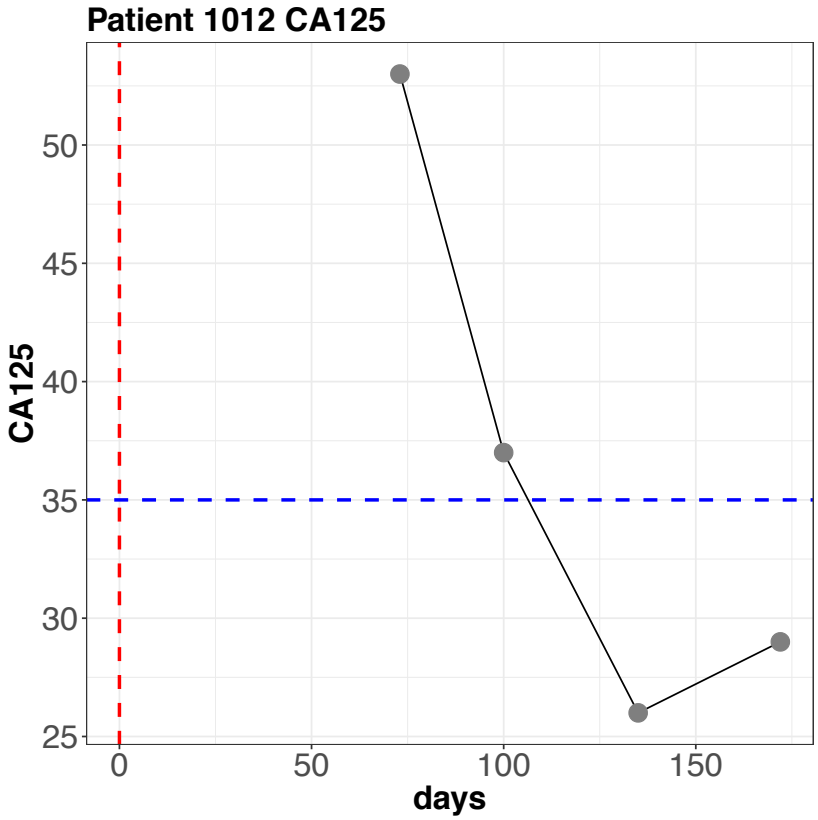

Patient 588 CA125
CA125
0
20
40
60
0
30
60
90
days

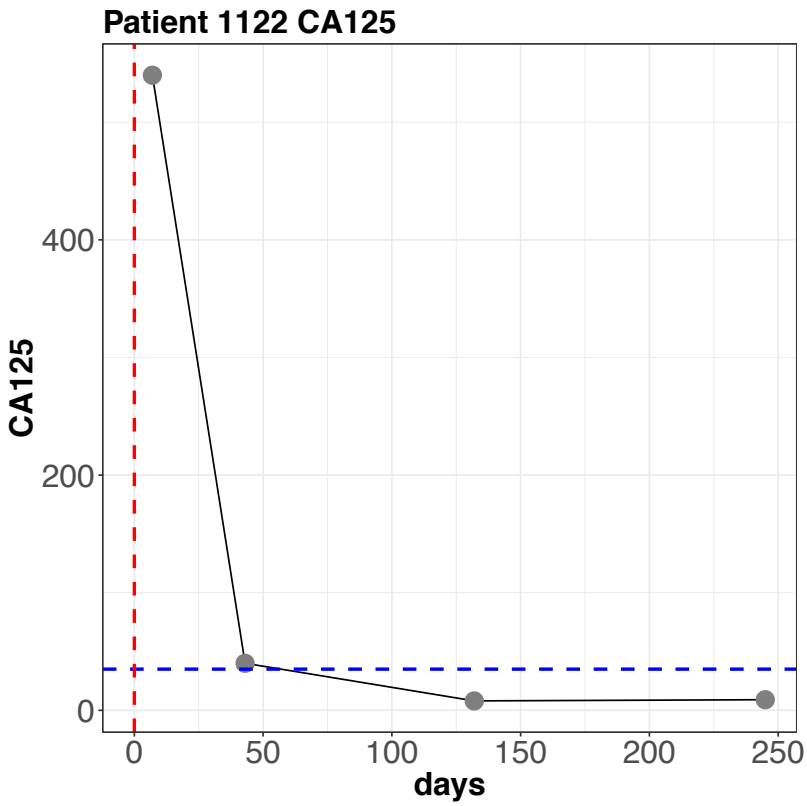

Patient 617 CA125
CA125
10
20
30
40
50
0
20
40
60
80
days

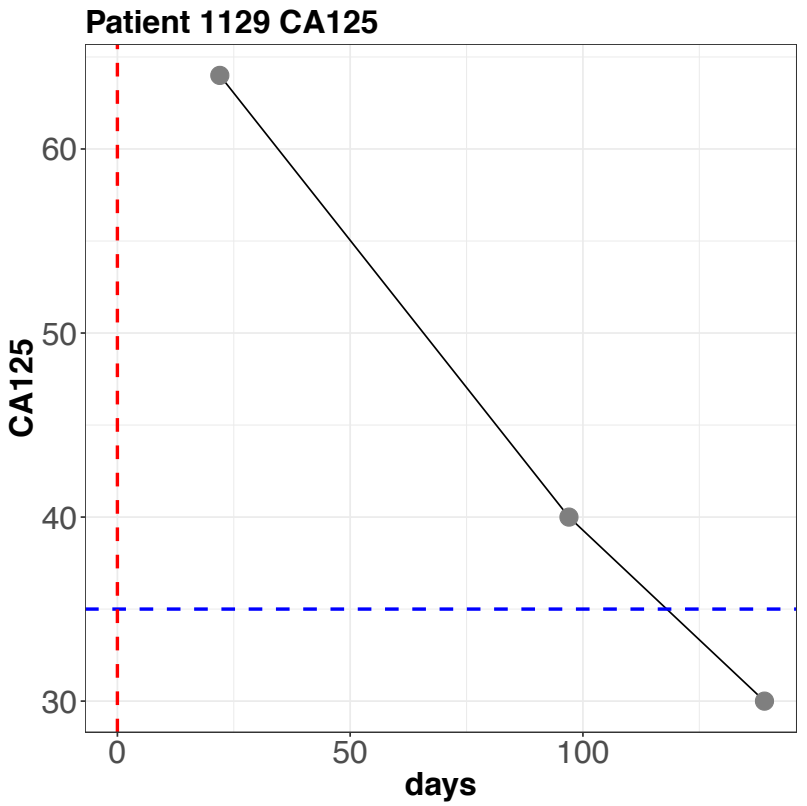

Patient 620 CA125
CA125
100
50
0
0
50
100
days

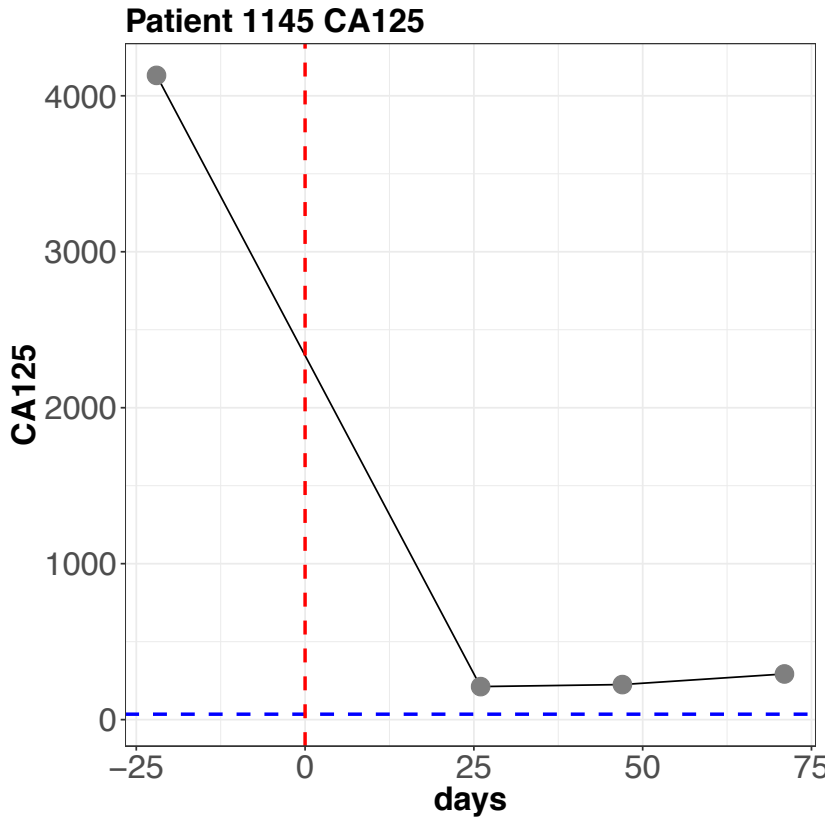

Patient 813 CA125
CA125
400
200
0
−40
0
40
days

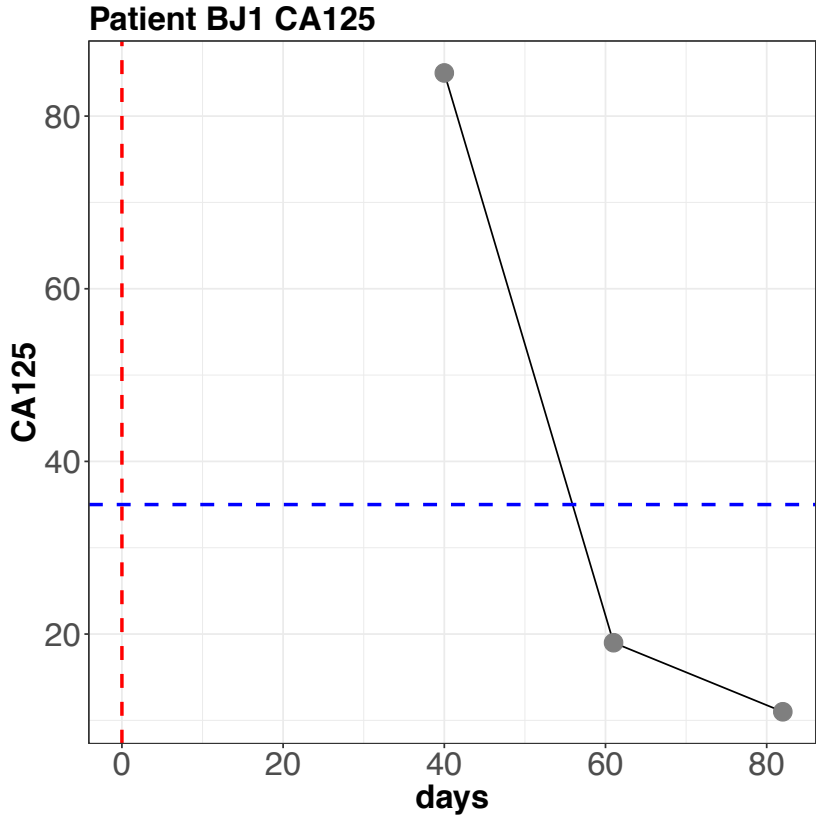

Patient 992 CA125
12000
9000
6000
3000
0
CA125
0
20
40
60
days


Patient 1012 CA125
50
45
40
35
30
25
CA125
0
50
100
150
days


Patient 1122 CA125
CA125
400
200
0
0
50
100
150
200
250
days


Patient 1129 CA125
CA125
60
50
40
30
0
50
100
days


Patient 1145 CA125
4000
3000
2000
1000
0
CA125
−25
0
25
50
75
days


Patient BJ1 CA125
80
60
40
20
CA125
0
20
40
60
80
days

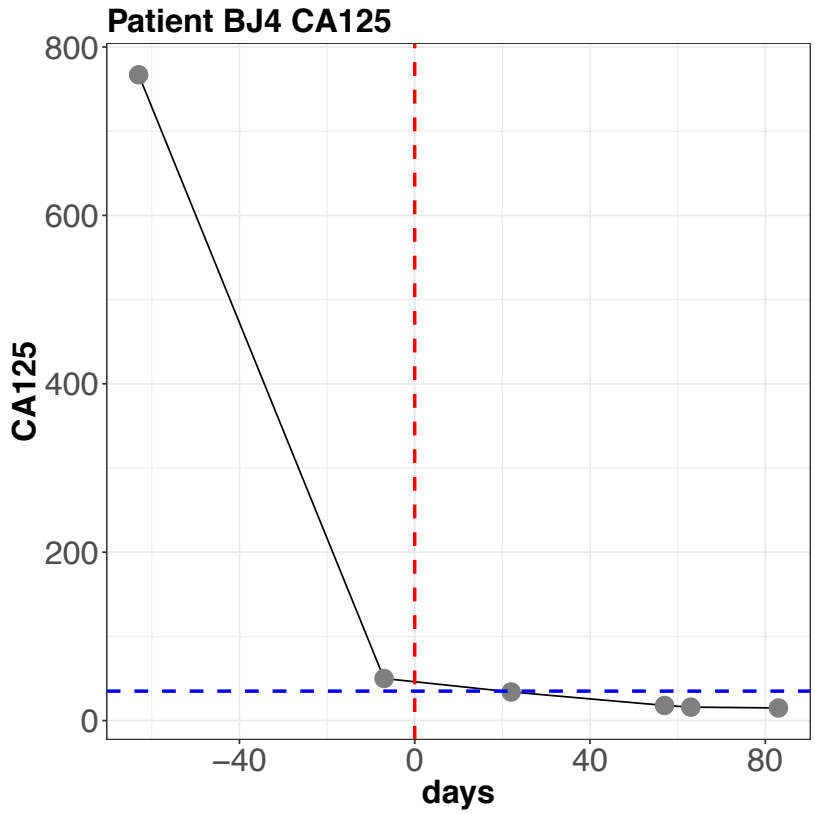


Figure 36 – Patients are considered to be responsive to treatments if their respective CA-125 values dropped below normal values (<35) within 60 days of the start of chemotherapeutic treatment (red dashed line indicates date of surgery).

REFERENCES

1. Popescu, B., et al., *NGS combined with phylogenetic analysis to detect HIV-1 dual infection in Romanian people who inject drugs.* Microbes Infect, 2018.

2. Chen, L., et al., *Correlation between RNA-Seq and microarrays results using TCGA data.* Gene, 2017. **628**: p. 200-204.

3. Perkel, J.M., *How bioinformatics tools are bringing genetic analysis to the masses.* Nature, 2017. **543**(7643): p. 137-138.

4. Sharp, C., et al., *Oxford Screening CSF and Respiratory samples ('OSCAR'): results of a pilot study to screen clinical samples from a diagnostic microbiology laboratory for viruses using Illumina next generation sequencing.* BMC Res Notes, 2018. **11**(1): p. 120.

5. Chen, X.W. and J.X. Gao, *Big Data Bioinformatics.* Methods, 2016. **111**: p. 1-2.

6. Sanguinetti, G., et al., *Accounting for probe-level noise in principal component analysis of microarray data.* Bioinformatics, 2005. **21**(19): p. 3748-54.

7. Boareto, M. and N. Caticha, *t-Test at the Probe Level: An Alternative Method to Identify Statistically Significant Genes for Microarray Data.* Microarrays (Basel), 2014. **3**(4): p. 340-51.

8. Arloth, J., et al., *Re-Annotator: Annotation Pipeline for Microarray Probe Sequences.* PLoS One, 2015. **10**(10): p. e0139516.

9. Uziela, K. and A. Honkela, *Probe Region Expression Estimation for RNA-Seq Data for Improved Microarray Comparability.* PLoS One, 2015. **10**(5): p. e0126545.

10. Costello, J.C., et al., *A community effort to assess and improve drug sensitivity prediction algorithms.* Nat Biotech, 2014. **32**(12): p. 1202-1212.

11. Geeleher, P., N. Cox, and R.S. Huang, *pRRophetic: an R package for prediction of clinical chemotherapeutic response from tumor gene expression levels.* PLoS One, 2014. **9**(9): p. e107468.

12. Nitsche, B.M., A.F. Ram, and V. Meyer, *The use of open source bioinformatics tools to dissect transcriptomic data.* Methods Mol Biol, 2012. **835**: p. 311-31.

13. Harris, N.L., et al., *The 2017 Bioinformatics Open Source Conference (BOSC).* F1000Res, 2017. **6**.

14. Aymard, F., et al., *Genome-wide mapping of long-range contacts unveils clustering of DNA double-strand breaks at damaged active genes.* Nat Struct Mol Biol, 2017. **24**(4): p. 353-361.

15. Joensen, K.G., et al., *Whole-genome sequencing of Campylobacter jejuni isolated from Danish routine human stool samples reveals surprising degree of clustering.* Clin Microbiol Infect, 2018. **24**(2): p. 201 e5-201 e8.

16. Okada, D., F. Ino, and K. Hagihara, *Accelerating the Smith-Waterman algorithm with interpair pruning and band optimization for the all-pairs comparison of base sequences.* BMC Bioinformatics, 2015. **16**: p. 321.

17. Xu, Z., Y. Yang, and B. Huang, *A teaching approach from the exhaustive search method to the Needleman-Wunsch algorithm.* Biochem Mol Biol Educ, 2017. **45**(3): p. 194-204.

18. Sun, J., et al., *Multiple Sequence Alignment with Hidden Markov Models Learned by Random Drift Particle Swarm Optimization.* IEEE/ACM Trans Comput Biol Bioinform, 2014. **11**(1): p. 243-57.

19. Audano, P. and F. Vannberg, *KAnalyze: a fast versatile pipelined k-mer toolkit.* Bioinformatics, 2014. **30**(14): p. 2070-2.

20. Claussen, J.C., et al., *Boolean analysis reveals systematic interactions among low-abundance species in the human gut microbiome.* PLoS Comput Biol, 2017. **13**(6): p. e1005361.

21. Inza, I., et al., *Machine learning: an indispensable tool in bioinformatics.* Methods Mol Biol, 2010. **593**: p. 25-48.

22. Olson, R.S., et al., *Data-driven advice for applying machine learning to bioinformatics problems.* Pac Symp Biocomput, 2018. **23**: p. 192-203.

23. Bahcall, O., *Precision medicine.* Nature, 2015. **526**(7573): p. 335.

24. Abrahams, E. and S.L. Eck, *Molecular medicine: Precision oncology is not an illusion.* Nature, 2016. **539**(7629): p. 357.

25. Eyal-Altman, N., M. Last, and E. Rubin, *PCM-SABRE: a platform for benchmarking and comparing outcome prediction methods in precision cancer medicine.* BMC Bioinformatics, 2017. **18**(1): p. 40.

26. Letai, A., *Functional precision cancer medicine-moving beyond pure genomics.* Nat Med, 2017. **23**(9): p. 1028-1035.

27. Wang, C.W., et al., *A Benchmark for Comparing Precision Medicine Methods in Thyroid Cancer Diagnosis using Tissue Microarrays.* Bioinformatics, 2017.

28. Xu, C., et al., *Functional precision medicine identifies novel druggable targets and therapeutic options in head and neck cancer.* Clin Cancer Res, 2018.

29. Lee, C., A. Abdool, and C.H. Huang, *PCA-based population structure inference with generic clustering algorithms.* BMC Bioinformatics, 2009. **10 Suppl 1**: p. S73.

30. Gysels, E., P. Renevey, and P. Celka, *SVM-based recursive feature elimination to compare phase synchronization computed from broadband and narrowband EEG signals in brain-computer interfaces.* Signal Processing, 2005. **85**(11): p. 2178-2189.

31. Khanna, S., A. Sattar, and D. Hansen, *Advances in artificial intelligence research in health.* Australas Med J, 2012. **5**(9): p. 475-7.

32. Liu, T., et al., *Prediction of subcellular location of apoptosis proteins combining tri-gram encoding based on PSSM and recursive feature elimination.* Journal of theoretical biology, 2015. **366**: p. 8-12.

33. Bolstad, B.M., et al., *A comparison of normalization methods for high density oligonucleotide array data based on variance and bias.* Bioinformatics, 2003. **19**(2): p. 185-193.

34. Amaratunga, D. and J. Cabrera, *Analysis of data from viral DNA microchips.* Journal of the American Statistical Association, 2001. **96**(456): p. 1161-1170.

35. Pruitt, K.D., T. Tatusova, and D.R. Maglott, *NCBI reference sequences (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins.* Nucleic Acids Res, 2007. **35**(Database issue): p. D61-5.

36. Salichos, L. and A. Rokas, *Inferring ancient divergences requires genes with strong phylogenetic signals.* Nature, 2013. **497**(7449): p. 327-31.

37. Altschul, S.F., et al., *Basic local alignment search tool.* J Mol Biol, 1990. **215**(3): p. 403-10.

38. Vingron, M. and M.S. Waterman, *Sequence alignment and penalty choice. Review of concepts, case studies and implications.* J Mol Biol, 1994. **235**(1): p. 1-12.

39. Morgenstern, B., A. Dress, and T. Werner, *Multiple DNA and protein sequence alignment based on segment-to-segment comparison.* Proc Natl Acad Sci U S A, 1996. **93**(22): p. 12098-103.

40. Bawono, P., et al., *Multiple Sequence Alignment.* Methods Mol Biol, 2017. **1525**: p. 167-189.

41. Ewans, L.J., et al., *Whole-exome sequencing reanalysis at 12 months boosts diagnosis and is cost-effective when applied early in Mendelian disorders.* Genet Med, 2018.

42. Zielezinski, A., et al., *Alignment-free sequence comparison: benefits, applications, and tools.* Genome Biol, 2017. **18**(1): p. 186.

43. Chor, B., et al., *Genomic DNA k-mer spectra: models and modalities.* Genome Biol, 2009. **10**(10): p. R108.

44. Doolittle, R.F., *Biodiversity: microbial genomes multiply.* Nature, 2002. **416**(6882): p. 697-700.

45. Fu, S., A.M. Tarone, and S.H. Sze, *Heuristic pairwise alignment of de Bruijn graphs to facilitate simultaneous transcript discovery in related organisms from RNA-Seq data.* BMC Genomics, 2015. **16 Suppl 11**: p. S5.

46. Compeau, P.E., P.A. Pevzner, and G. Tesler, *How to apply de Bruijn graphs to genome assembly.* Nat Biotechnol, 2011. **29**(11): p. 987-91.

47. Novak, A.M., E. Garrison, and B. Paten, *A graph extension of the positional Burrows-Wheeler transform and its applications.* Algorithms Mol Biol, 2017. **12**: p. 18.

48. Li, X., et al., *Detecting Esophageal Cancer Using Surface-Enhanced Raman Spectroscopy (SERS) of Serum Coupled with Hierarchical Cluster Analysis and Principal Component Analysis.* Appl Spectrosc, 2015. **69**(11): p. 1334-41.

49. Campo, D.S., et al., *Accurate Genetic Detection of Hepatitis C Virus Transmissions in Outbreak Settings.* J Infect Dis, 2016. **213**(6): p. 957-65.

50. Glebova, O., et al., *Inference of genetic relatedness between viral quasispecies from sequencing data.* BMC Genomics, 2017. **18**(Suppl 10): p. 918.

51. Skums, P., et al., *QUENTIN: reconstruction of disease transmissions from viral quasispecies genomic data.* Bioinformatics, 2018. **34**(1): p. 163-170.

52. Bousema, T., et al., *Asymptomatic malaria infections: detectability, transmissibility and public health relevance.* Nat Rev Microbiol, 2014. **12**(12): p. 833-40.

53. Ansari, H.R., et al., *Genome-scale comparison of expanded gene families in Plasmodium ovale wallikeri and Plasmodium ovale curtisi with Plasmodium malariae and with other Plasmodium species.* Int J Parasitol, 2016. **46**(11): p. 685-96.

54. Sutherland, C.J., et al., *Two nonrecombining sympatric forms of the human malaria parasite Plasmodium ovale occur globally.* J Infect Dis, 2010. **201**(10): p. 1544-50.

55. Soledad-Cadona, J., et al., *Pathogenicity Islands Distribution in Non-O157 Shiga Toxin-Producing Escherichia coli (STEC).* Genes (Basel), 2018. **9**(2).

56. Jaros, P., et al., *PFGE for Shiga toxin-producing Escherichia coli O157:H7 (STEC O157) and non-O157 STEC.* Methods Mol Biol, 2015. **1301**: p. 171-89.

57. Ju, W., et al., *Phylogenetic analysis of non-O157 Shiga toxin-producing Escherichia coli strains by whole-genome sequencing.* J Clin Microbiol, 2012. **50**(12): p. 4123-7.

58. Haverty, P.M., et al., *Reproducible pharmacogenomic profiling of cancer cell line panels.* Nature, 2016. **533**(7603): p. 333-7.

59. Haibe-Kains, B., et al., *Inconsistency in large pharmacogenomic studies.* Nature, 2013. **504**(7480): p. 389-93.

60. Piccolo, S.R., et al., *Multiplatform single-sample estimates of transcriptional activation.* Proc Natl Acad Sci U S A, 2013. **110**(44): p. 17778-83.

61. Esteva, A., et al., *Dermatologist-level classification of skin cancer with deep neural networks.* Nature, 2017. **542**(7639): p. 115-118.

62. Russakovsky, O., et al., *ImageNet Large Scale Visual Recognition Challenge.* International Journal of Computer Vision, 2015. **115**(3): p. 211-252.

63. Clough, E. and T. Barrett, *The Gene Expression Omnibus Database.* Methods Mol Biol, 2016. **1418**: p. 93-110.

64. Liu, B., et al., *Combining evolutionary information extracted from frequency profiles with sequence-based kernels for protein remote homology detection.* Bioinformatics, 2014. **30**(4): p. 472-9.

65. Hsu, C.-W., et al., *A practical guide to support vector classification.* 2003.

66. Saeys, Y., I. Inza, and P. Larrañaga, *A review of feature selection techniques in bioinformatics.* bioinformatics, 2007. **23**(19): p. 2507-2517.

67. Radovic, M., et al., *Minimum redundancy maximum relevance feature selection approach for temporal gene expression data.* BMC Bioinformatics, 2017. **18**(1): p. 9.

68. Gaul, D.A., et al., *Highly-accurate metabolomic detection of early-stage ovarian cancer.* Sci Rep, 2015. **5**: p. 16351.

69. Guan, W., et al., *Ovarian cancer detection from metabolomic liquid chromatography/mass spectrometry data by support vector machines.* BMC Bioinformatics, 2009. **10**: p. 259-259.

70. Wen, Y., et al., *Chemotherapeutic-induced apoptosis: a phenotype for pharmacogenomics studies.* Pharmacogenet Genomics, 2011. **21**(8): p. 476-88.

71. Consortium, T.I.C.G., *International network of cancer genome projects.* Nature, 2010. **464**: p. 993-998.

72. Hoadley, K.A., et al., *Multiplatform analysis of 12 cancer types reveals molecular classification within and across tissues of origin.* Cell, 2014. **158**(4): p. 929-944.

73. Azuaje, F., *Computational models for predicting drug responses in cancer research.* Brief Bioinform, 2017. **18**(5): p. 820-829.

74. Frampton, G.M., et al., *Development and validation of a clinical cancer genomic profiling test based on massively parallel DNA sequencing.* Nat Biotechnol, 2013. **31**(11): p. 1023-31.

75. Ozols, R.F., *Challenges for chemotherapy in ovarian cancer.* Ann Oncol, 2006. **17 Suppl 5**: p. v181-7.

76. Hansen, S., et al., *Gemcitabine in the treatment of ovarian cancer.* Annals of oncology, 1999. **10**: p. 51-54.

77. Bode, A.M. and Z. Dong, *Precision oncology- the future of personalized cancer medicine?* npj Precision Oncology, 2017. **1**(1): p. 2.

78. Salesse, S. and C.M. Verfaillie, *BCR/ABL: from molecular mechanisms of leukemia induction to treatment of chronic myelogenous leukemia.* Oncogene, 2002. **21**(56): p. 8547-59.

79. Saez-Rodriguez, J., et al., *Crowdsourcing biomedical research: leveraging communities as innovation engines.* Nat Rev Genet, 2016. **17**(8): p. 470-486.

80. Prasad, V., T. Fojo, and M. Brada, *Precision oncology: origins, optimism, and potential.* Lancet Oncol, 2016. **17**(2): p. e81-e86.

81. Druker, B.J., et al., *Effects of a selective inhibitor of the Abl tyrosine kinase on the growth of Bcr-Abl positive cells.* Nat Med, 1996. **2**(5): p. 561-6.

82. Tu, S.M., M.A. Bilen, and N.M. Tannir, *Personalised cancer care: promises and challenges of targeted therapy.* J R Soc Med, 2016. **109**(3): p. 98-105.

83. Hilbert, M. and P. Lopez, *The world's technological capacity to store, communicate, and compute information.* Science, 2011. **332**(6025): p. 60-5.

84. Vidyasagar, M., *Identifying predictive features in drug response using machine learning: opportunities and challenges.* Annu Rev Pharmacol Toxicol, 2015. **55**: p. 15-34.

85. Huang, C., et al., *Open source machine-learning algorithms for the prediction of optimal cancer drug therapies.* PLoS One, 2017. **12**(10): p. e0186906.

86. Saeys, Y., I. Inza, and P. Larranaga, *A review of feature selection techniques in bioinformatics.* Bioinformatics, 2007. **23**(19): p. 2507-17.

87. Lili, L.N., et al., *Molecular profiling supports the role of epithelial-to-mesenchymal transition (EMT) in ovarian cancer metastasis.* J Ovarian Res, 2013. **6**(1): p. 49.

88. Rustin, G.J., et al., *Definitions for response and progression in ovarian cancer clinical trials incorporating RECIST 1.1 and CA 125 agreed by the Gynecological Cancer Intergroup (GCIG).* Int J Gynecol Cancer, 2011. **21**(2): p. 419-23.

89. Gottesman, M.M., et al., *Toward a Better Understanding of the Complexity of Cancer Drug Resistance.* Annu Rev Pharmacol Toxicol, 2016. **56**: p. 85-102.

90. McDonald, J.F., *Back to the future - The integration of big data with machine learning is re-establishing the importance of predictive correlations in ovarian cancer diagnostics and therapeutics.* Gynecol Oncol, 2018.

91. Reuben, A., et al., *Genomic and immune heterogeneity are associated with differential responses to therapy in melanoma.* NPJ Genom Med, 2017. **2**.

92. Gordinier, M.E., et al., *Thiotepa in combination with cisplatin for primary epithelial ovarian cancer: a phase II study.* Int J Gynecol Cancer, 2002. **12**(6): p. 710-4.

93. Ma, X., et al., *Targeting CD146 in combination with vorinostat for the treatment of ovarian cancer cells.* Oncol Lett, 2017. **13**(3): p. 1681-1687.

94. Xiang, X., et al., *Sulfolobus tengchongensis spindle-shaped virus STSV1: virus-host interactions and genomic features.* J Virol, 2005. **79**(14): p. 8677-86.

95. Truncaite, L., et al., *Bacteriophage vB_EcoM_FV3: a new member of "rV5-like viruses".* Arch Virol, 2012. **157**(12): p. 2431-5.

96. Han, K.G., S.S. Lee, and C. Kang, *Soluble expression of cloned phage K11 RNA polymerase gene in Escherichia coli at a low temperature.* Protein Expr Purif, 1999. **16**(1): p. 103-8.

97. Haring, M., et al., *Viral diversity in hot springs of Pozzuoli, Italy, and characterization of a unique archaeal virus, Acidianus bottle-shaped virus, from a new family, the Ampullaviridae.* J Virol, 2005. **79**(15): p. 9904-11.

98. Gumerov, V.M., et al., *Complete genome sequence of "Vulcanisaeta moutnovskia" strain 768-28, a novel member of the hyperthermophilic crenarchaeal genus Vulcanisaeta.* J Bacteriol, 2011. **193**(9): p. 2355-6.